



# **NAVAL POSTGRADUATE SCHOOL**

**MONTEREY, CALIFORNIA**

## **THESIS**

**TOMAHAWK LAND ATTACK MISSILE  
PREDESIGNATION:  
OPTIMIZATION REVISITED**

by

Ali Demir

June 2004

Thesis Advisor:  
Second Reader:

Javier Salmeron  
Richard E. Rosenthal

**Approved for public release; distribution is unlimited**

THIS PAGE INTENTIONALLY LEFT BLANK

<b>REPORT DOCUMENTATION PAGE</b>			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
<b>1. AGENCY USE ONLY (Leave blank)</b>		<b>2. REPORT DATE</b> June 2004	<b>3. REPORT TYPE AND DATES COVERED</b> Master's Thesis	
<b>4. TITLE AND SUBTITLE:</b> Tomahawk Land Attack Missile Predesignation: Optimization Model And Algorithm Revisited			<b>5. FUNDING NUMBERS</b>	
<b>6. AUTHOR(S)</b> Demir, Ali				
<b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b> Naval Postgraduate School Monterey, CA 93943-5000			<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b>	
<b>9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b> Naval Surface Warfare Center Dahlgren Division (NCWSDD)			<b>10. SPONSORING/MONITORING AGENCY REPORT NUMBER</b>	
<b>11. SUPPLEMENTARY NOTES</b> The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
<b>12a. DISTRIBUTION / AVAILABILITY STATEMENT</b> Approved for public release; distribution is unlimited			<b>12b. DISTRIBUTION CODE</b>	
<b>13. ABSTRACT (maximum 200 words)</b> <p>The Tomahawk Land-Attack Missile (TLAM) is the long-range precision weapon of choice in strike warfare against strategic targets for U.S. military forces. Predesignation is the process of determining which ship or submarine will fire which TLAM missiles in support of an authorized attack upon specified targets. This thesis revisits the mathematical models and algorithms developed by previous NPS faculty and students to optimally conduct the allocation of TLAMs to firing units. We incorporate all the problem specifications addressed by previous heuristic algorithms for the problem, and compare our results to those in publicly available test cases. We show that our models can be solved optimally in affordable time for most of the cases and make provisions to establish accurate bounds in the other cases.</p>				
<b>14. SUBJECT TERMS</b> Tomahawk Land-Attack Missile, Long-Range Precision Weapon, Predesignation, Optimization, Multi-Objective Programming.			<b>15. NUMBER OF PAGES</b> 106	
			<b>16. PRICE CODE</b>	
<b>17. SECURITY CLASSIFICATION OF REPORT</b> Unclassified	<b>18. SECURITY CLASSIFICATION OF THIS PAGE</b> Unclassified	<b>19. SECURITY CLASSIFICATION OF ABSTRACT</b> Unclassified	<b>20. LIMITATION OF ABSTRACT</b> UL	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)  
Prescribed by ANSI Std. Z39-18

THIS PAGE INTENTIONALLY LEFT BLANK

**Approved for public release; distribution is unlimited**

**TOMAHAWK LAND ATTACK MISSILE PREDESIGNATION:  
OPTIMIZATION REVISITED**

Ali Demir  
First Lieutenant, Turkish Army  
B.S., Turkish Army Academy, 1997

Submitted in partial fulfillment of the  
requirements for the degree of

**MASTER OF SCIENCE IN OPERATIONS RESEARCH**

from the

**NAVAL POSTGRADUATE SCHOOL  
June 2004**

Author: Ali Demir

Approved by: Javier Salmeron  
Thesis Advisor

Richard E. Rosenthal  
Second Reader

James N. Eagle  
Chairman, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

## **ABSTRACT**

The Tomahawk Land-Attack Missile (TLAM) is the long-range precision weapon of choice in strike warfare against strategic targets for U.S. military forces. Predesignation is the process of determining which ship or submarine will fire which TLAM missiles in support of an authorized attack upon specified targets. This thesis revisits the mathematical models and algorithms developed by previous NPS faculty and students to optimally conduct the allocation of TLAMs to firing units. We incorporate all the problem specifications addressed by previous heuristic algorithms for the problem, and compare our results to those in publicly available test cases. We show that our models can be solved optimally in affordable time for most of the cases and make provisions to establish accurate bounds in the other cases.

THIS PAGE INTENTIONALLY LEFT BLANK



# TABLE OF CONTENTS

<b>I.</b>	<b>INTRODUCTION.....</b>	<b>1</b>
<b>A.</b>	<b>BACKGROUND .....</b>	<b>1</b>
	1. History.....	1
	2. Characteristics of Tomahawk Land Attack Missile .....	1
<b>B.</b>	<b>FIRING UNITS.....</b>	<b>3</b>
	1. Ships .....	3
	2. Submarines .....	6
<b>C.</b>	<b>PREDESIGNATION .....</b>	<b>7</b>
<b>D.</b>	<b>STRIKE PLANNING CONSIDERATIONS.....</b>	<b>8</b>
	1. General Considerations .....	8
	2. Near-Simultaneous Tasks.....	9
	3. Missile Redundancy .....	11
	4. Other Considerations.....	12
<b>E.</b>	<b>SUMMARY OF PAST WORK .....</b>	<b>13</b>
<b>F.</b>	<b>THESIS OUTLINE.....</b>	<b>15</b>
<b>II.</b>	<b>MIXED-INTEGER LINEAR OPTIMIZATION MODEL .....</b>	<b>17</b>
<b>A.</b>	<b>PROBLEM SPECIFICATIONS: DEFINITIONS AND ASSUMPTIONS.....</b>	<b>17</b>
	1. Near-Simultaneous Tasks.....	17
	2. Firing Units.....	17
	3. Ships .....	18
	4. Submarines .....	18
	5. Targets .....	18
	6. Tasks, Task-Parts and Task-Part Missiles .....	18
	7. Launch Areas .....	20
	8. Weapons.....	21
	9. Tomahawk Strike Coordinator Allocation.....	21
	10. Goal Specifications.....	21
	a. Goal #1.....	21
	b. Goal #2.....	22
	c. Goal #3.....	22
	d. Goal #4.....	22
	e. Goal #5.....	22
	f. Goal #6.....	22
	g. Goal #7.....	22
	h. Goal #8.....	23
<b>B.</b>	<b>TLAM ALLOCATION MODEL .....</b>	<b>23</b>
	1. Basic Model with Goal #1 .....	23
	2. Goal #2 .....	26
	3. Goal #3 .....	27
	4. Goal #4 .....	27
	5. Goal #5 .....	28

6.	Goal #6 .....	29
7.	Goal #7 .....	30
8.	Goal #8 .....	30
9.	Restricted Targets.....	31
10.	Non-separable Tasking.....	31
11.	TSC Specifications .....	32
12.	Submarines .....	32
13.	Levels of Achievement for Goals .....	33
14.	Combined Objective Function.....	33
15.	Multi-Objective Formulation.....	34
C.	PRELIMINARY VALIDATION OF THE MODEL .....	34
D.	SOLVING TLAM ALLOCATION MODEL USING FIX-AND-RELAX.....	37
III.	IMPLEMENTATION AND COMPUTATIONAL RESULTS.....	43
A.	OVERVIEW.....	43
1.	Strategy File.....	43
2.	Goal Parameters File .....	46
3.	Output Files .....	47
4.	Example .....	50
B.	SCENARIO DESCRIPTION AND COMPUTATIONAL RESULTS .....	51
1.	Scenario Description.....	51
2.	Test Case Set Up.....	52
3.	Results .....	53
4.	Refinements to Tackle the Hardest Test Cases .....	56
a.	<i>Reducing the Solving Time.....</i>	57
b.	<i>Using Fix and Relax .....</i>	57
IV.	CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER RESEARCH .....	59
	APPENDIX A. FULL-TLAM ALLOCATION MODEL.....	61
A.	NOTATION.....	61
1.	Indices .....	61
2.	Sets.....	61
3.	Data .....	62
4.	Derived Data and Sets .....	63
5.	Decision Variables.....	63
B.	OBJECTIVE FUNCTION .....	65
C.	MODEL CONSTRAINTS.....	66
1.	Constraint Formulation.....	66
2.	Multi-Objective Formulation.....	73
3.	Description of Formulation.....	73
	LIST OF REFERENCES.....	77
	INITIAL DISTRIBUTION LIST .....	79

## LIST OF FIGURES

Figure 1.	Tomahawk Land Attack Missile.....	2
Figure 2.	TLAM Firing Units.....	3
Figure 3.	Standard and Crane Modules.....	4
Figure 4.	Full-Size and Half-Size VLS Layouts. ....	5
Figure 5.	Cutaway View of a Torpedo Tube Room.....	7
Figure 6.	Capsule Launching System .....	7
Figure 7.	Conflicting Sets of Tasks.....	10
Figure 8.	Module with an Empty Half-Module and Full Half-Module.....	11
Figure 9.	Separable and Non-separable Tasking.....	12
Figure 10.	Firing Units.....	17
Figure 11.	Tasking Levels.....	20
Figure 12.	Task Specifications for Validation Case.....	35
Figure 13.	Firing Units for Validation Case.....	35
Figure 14.	Non-Optimal Solution of Validation Case.....	35
Figure 15.	Optimal Solution of Validation Case Obtained with the Hierarchical Formulation.....	36
Figure 16.	Near-Optimal Solution of Validation Case Obtained with the Combined- Goal Formulation.....	36
Figure 17.	Goal Achievements for Validation Case.....	37
Figure 18.	Strategy Data File.....	44
Figure 19.	Goal Parameters Data File.....	46
Figure 20.	Goal Achievement Results: Combined-Goal Formulation.....	48
Figure 21.	Goal Achievement Results: Hierarchical Formulation.....	48
Figure 22.	Allocation Results.....	49
Figure 23.	Case 1: Changing Goal Priorities.....	50
Figure 24.	Case 5A: Comparison of Solution Values and Best Bounds.....	51
Figure 25.	Case 5B: Reducing Maximum Time.....	57

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF TABLES

Table 1.	Result List .....	56
----------	-------------------	----

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF TERMS AND ABBREVIATIONS

Remark: The majority of the following terms have been adopted from Arnold [2000].

<b>Aim-point</b>	A two-letter designation used to identify subdivisions of a <i>target</i> . These subdivisions are used to refine impact points for <i>TLAMs</i> .
<b>Alignment</b>	The process of preparing a <i>TLAM</i> for launch. During alignment, the missile's navigational equipment is energized, the onboard computers are activated, and the software and data are loaded.
<b>Back-up Task-Part</b>	A task-part assigned to a firing unit in order to provide redundancy for a primary task-part of the same <i>task</i> . The firing unit will not fire a back-up task-part unless its associated primary task-part and ready-spare task-part fail.
<b>Basic Encyclopedia Number (BEN)</b>	An 11-digit alphanumeric code used to designate a target site.
<b>Block</b>	The software and engine combination of a <i>TLAM</i> .
<b>Canister Capsule</b>	The component within <i>CLS</i> of a submarine that stores <i>TLAMs</i> .
<b>Capsule Launching</b>	A submarine <i>TLAM</i> launching system that provides a System (CLS) specific number of canister capsules exclusively for launching <i>TLAMs</i> .
<b>Cell</b>	The component within a <i>VLS</i> of a ship that stores missiles, including <i>TLAMs</i> .
<b>Digital Scene Matching Area Correlation (DSMAC)</b>	An electro-optical sensor system that takes images from the ground below the missile and compares these to reference images stored in the on-board computer.

<b>Expend Firing Unit</b>	The <i>firing units</i> that are departing the theater of operations very soon or entering port for a period of extended maintenance.
<b>Firing Unit</b>	A ship or submarine that can be assigned a <i>TLAM task</i> .
<b>Geo-Feasible</b>	Yes/No characteristic for the <i>firing units</i> that establishes whether it can fire <i>TLAMs</i> at a particular target or not.
<b>Half-module</b>	A four-cell component of a <i>module</i> within the <i>VLS</i> .
<b>Inertial Navigation System (INS)</b>	A cluster of sensors that tracks the position and orientation of a body to which the cluster is rigidly attached.
<b>Ghost Task-Part</b>	A <i>task-part</i> that simultaneously provides redundancy for multiple <i>tasks</i> and task-parts.
<b>Global Positioning System (GPS)</b>	A navigational and positioning system developed by the U.S. Department of Defense, by which the location of a position on or above the Earth can be determined by a special receiver at that point interpreting signals received simultaneously from several of a constellation of special satellites.
<b>Launcher</b>	The combination of <i>VLS modules</i> onboard a ship.
<b>Launch Area</b>	For a <i>target</i> a large geo-graphic region from which a TLAM can be fired and can reach the target.
<b>Missile Mission Matching (M<sup>3</sup>) List</b>	A prioritized list of TLAMs capable of fulfilling a <i>task</i> .
<b>Mission</b>	A set of three-dimensional coordinates that designates an over-land flight path from a <i>TLAM's</i> point of launch to the <i>target</i> site. A mission defines the target, aim-points, warhead, block, and launch area.



<b>Module</b>	The basic component of a <i>VLS</i> , consisting of eight cells.
<b>Non-separable Task-part</b>	A <i>task-part</i> that in which all the missiles must be allocated to the same <i>firing unit</i> .
<b>Naval Surface Warfare Center Dahlgren Division (NSWCDD)</b>	A naval command, located in Dahlgren, VA, that conducts research and development on weapon systems.
<b>Predesignation</b>	The process used to determine which firing unit will fire <i>TLAMs</i> during an attack. Predesignation is conducted in two phases. In Phase 1, the <i>TSC</i> allocates <i>TLAM</i> target assignments to ships and submarines. In Phase 2, these assignments are allocated to specific <i>TLAMs</i> onboard individual firing unit based on additional considerations not accounted in Phase 1.
<b>Primary Task-Part</b>	A <i>task-part</i> required by a <i>firing unit</i> to prepare a missile to launch and to fire against a specific target during a specified <i>time period</i> .
<b>Ready-Spare Task-Part</b>	A <i>task-part</i> assigned to a <i>firing unit</i> in order to provide redundancy for a <i>primary task-part</i> of the same <i>task</i> .
<b>Restricted Target</b>	A <i>target</i> that (a) needs all <i>tasks</i> and <i>task-parts</i> be allocated to a limited number of <i>firing units</i> , given independently for each restricted target and (b) needs to allocate all (or none) of the tasks in the target.
<b>Salvo Size</b>	The maximum number of <i>TLAMs</i> that can simultaneously be <i>aligned</i> and launched from a <i>firing unit</i> .
<b>Statute Mile</b>	A unit of length equal to 1,760 yards land mile.
<b>Strike</b>	An attack of <i>TLAM</i> consisting of multiple, overlapping <i>time periods</i> .

<b>Target (I)</b>	An <i>aim-point</i> on the geo-graphical area that is to be fired from ships or submarines.
<b>Target (II)</b>	A set of one or more <i>tasks</i> to be carried out. A target with a limit on the number of <i>firing units</i> to which its corresponding tasks can be assigned is called <i>restricted target</i> . In addition, restricted targets require all (or none) of their tasks to be allocated.
<b>Task</b>	The combination of <i>mission</i> and <i>time period</i> .
<b>Task-Part</b>	Components of a task that indicate whether a <i>TLAM</i> is to be fired at a <i>target</i> or <i>aligned</i> to provide redundancy. There are four types of task-parts. <i>Primary task-parts</i> are fired at the target while <i>ready-spare</i> , <i>back-up</i> and <i>ghost task-parts</i> provide redundancy for the primary task-parts.
<b>Task-Part Missile</b>	A missile on a <i>firing unit</i> , associated with a <i>task-part</i> , that the firing unit is directed to <i>align</i> and prepare for launch. A task-part may have one or more task-part missiles.
<b>Terrain Counter Matching (TERCOM)</b>	A method of terrain navigation in which the missile's position is derived by correlating a sensed terrain profile (synthesized from radar altimeter measurements) to a map terrain profile.
<b>Time Period</b>	An allowable interval of time during which a <i>TLAM</i> may be fired to complete a <i>mission</i> .
<b>Tomahawk Land Attack Missile (TLAM)</b>	An all-weather submarine- or ship-launched land-attack cruise missile.
<b>Tomahawk Strike Coordinator (TSC)</b>	The officer on the Battle Group or Naval Component Commander staff responsible for the employment of <i>TLAMs</i> .
<b>Torpedo Tube</b>	A system used onboard submarines to fire torpedoes or <i>TLAMs</i> .

**Vertical Launching System (VLS)**

A system onboard a surface ship used to store, prepare and launch TLAMs, surface-to-air missiles, and rocket-thrown torpedoes.

**Warhead**

The part of the missile, typically the head or tip, that gets loaded with whatever weaponry payload the missile is supposed to deliver.

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF ACRONYMS, SYMBOLS

ATL	Average Time to Launch
BEN	Basic Encyclopedia Number
B&B	Branch-and-Bound
CLS	Capsule Launching System
CNO	Chief of Naval Operations
CINC	Commander-in-Chief
FY	Fiscal Year
F&R	Fix-and-Relax
GPS	Global Positioning Satellite System
INS	Inertial Navigation System
IOC	Initial Operational Capability
M <sup>3</sup>	Missile Mission Matching
MIN	Mission Identification Number
MIP	Mixed-integer Programming
MO	Multi-objective
mph	Miles per Hour
nm	Nautical Mile
NCA	National Command Authority
NPS	Naval Postgraduate School
NSWCDD	Naval Surface Warfare Center Dahlgren Division
SSN	Navy Attack Submarine
TERCOM	Terrain Contour Matching
TLAM	Tomahawk Land Attack Missile
TSC	Tomahawk Strike Coordinator
VLS	Vertical Launching System

THIS PAGE INTENTIONALLY LEFT BLANK

## **ACKNOWLEDGMENTS**

I would like to acknowledge the assistance of my advisor Professor Javier Salmeron, thank him for his guidance, encouragement, support and patience, without which this research would not have been successful. His schedule, open door policy and help throughout my studies were instrumental in the achievement of this thesis. In addition, I would like to thank Professor Richard E. Rosenthal for his advice, insight and experience contributing to this thesis.

Finally, I would like to thank my wife, Fatma, for her patience and support throughout this process and for giving me encouragement whenever I needed.

THIS PAGE INTENTIONALLY LEFT BLANK



## EXECUTIVE SUMMARY

The Tomahawk Land-Attack Missile (TLAM) is the long-range accuracy strategic weapon of choice in strike warfare against strategic targets for United States (U.S.) military forces. When the U.S. National Command Authority, formed by the President and the Secretary of Defense, authorizes a TLAM strike, the authorization passes down through the chain of command via the Chief of Naval Operations, a regional fleet Commander-in-Chief, and ultimately to the Battle Group Tomahawk Strike Coordinator (TSC).

Predesignation is the process of determining which ship or submarine will fire which TLAM missiles in support of an authorized attack upon specified targets. Predesignation has two phases. In Phase 1, the TSC allocates targets to ships and submarines. In Phase 2, tasks are allocated to specific TLAMs onboard individual ships and submarines. Currently, the TLAM is mostly being used to attack a fixed stationary target such as a building, but under the conditions of modern warfare, the TLAM must be able to attack a mobile target on short notice. Our goal is to provide the TSC with an automatic predesignation procedure that enables tasks to be assigned quickly and efficiently.

Over the last five years, faculty and students at the Naval Postgraduate School (NPS) Operations Research Department have addressed this problem under the sponsorship of the Naval Surface Warfare Center, Dahlgren Division (NSWCDD). A series of student theses and other reports document this effort. As a result, a multi-objective (MO) mixed-integer programming (MIP) model was developed by the end of 1999. This model represented an important challenge in terms of solvability. In fact, previous work determined that real-size problems cannot be solved within reasonable optimality tolerance in affordable time. This difficulty was addressed in subsequent years by means of fast heuristic algorithms at the expense of solution quality and, to some extent, flexibility in the set up of goal priorities and other potential redesigns in the problem specifications. As the NPS team's knowledge of TLAM predesignation

evolved, the heuristics addressed features of the problem that were not considered in the earliest optimization models. The latest heuristic algorithm has been recently implemented by professional programmers at NSWCCD, adhering to algorithm specifications by NPS faculty. This thesis bridges the gap between the latest specifications implemented in the heuristic algorithms and the outdated mathematical formulation. We show that the revised formulation can be solved in reasonable time for the existing test cases, including those that incorporate recent features that were not incorporated in earlier versions of the formulation.

Specific accomplishments of this thesis are:

- Consolidation of all problem specifications to date.
- Incorporation of all specifications into a revised MO MIP formulation. In particular, the following considerations have been added to the optimization:
  - Submarines and their characteristics: torpedo tubes, missiles in the torpedo room, canister capsules, etc.
  - Target concept as a set of multiple tasks sharing certain joint constraints.
  - Multiple missiles might be required for every task-part.
  - TSC's manual allocation missiles to specific cells, torpedo tubes or canister capsules.
  - TSC specification regarding whether multiple missiles for a given task-part must be assigned to the same firing unit, or separating missiles across multiple firing units is allowed.
  - TSC specification regarding whether back-up task-parts can be allocated to submarines or not.
  - Incorporation of ghost task-parts and their characteristics.
- Exploration of alternative techniques to obtain accurate bounds on large-size test cases. We compare the results against those provided by the heuristic to provide empiric validation for using the heuristic as a reliable tool to support fast TLAM predesignation.

# **I. INTRODUCTION**

## **A. BACKGROUND**

### **1. History**

After the signing of first Strategic Arms Limitation Treaty, the strategic cruise missile was born. Until January 6, 1977, the United States (U.S.) Navy and U.S. Air Force maintained their own projects independently. On that date, the Defense Systems Acquisition Review Council authorized a full-scale engineering and testing program for air-launched, anti-surface, theater, and nuclear versions of the cruise missile. The Joint Cruise Missiles Project Office was formed to manage cruise missile development, and it became a joint U.S. Navy and Air Force project. Today, the U.S. Navy continues as a lead agency in cruise missile program management. [Reynolds, 1998.]

The first Tomahawk Land Attack Missile (TLAM) was deployed on board a Los Angeles class fast-attack submarine in November 1983. Its first operational use was during Operation Desert Storm in 1991. Since its first combat use, it has become the weapon of choice in strike warfare for U.S. Military forces. The governments of the U.S. and United Kingdom (U.K.) signed a Foreign Military Sales Agreement for the acquisition of 65 missiles in 1995, recorded as the first sale of TLAMs to a foreign country [Navy Office of Information, 2004.]. After a successful launch and live warhead test in November 1998, the U.K. declared operational capability. TLAMs can be currently launched from U.S. Navy surface ships and from U.S. Navy and Royal Navy submarines.

### **2. Characteristics of Tomahawk Land Attack Missile**

The TLAM (Figure 1) is a long range, subsonic cruise missile designed for land attack warfare. It was designed to hit strategic targets day and night with a minimum of collateral damage by flying extremely low altitudes at high subsonic speeds and piloting over an evasive route by several tailored mission-guidance systems. It is able to convey a 1,000 pound warhead to a range of approximately 1,000nm with pinpoint accuracy.

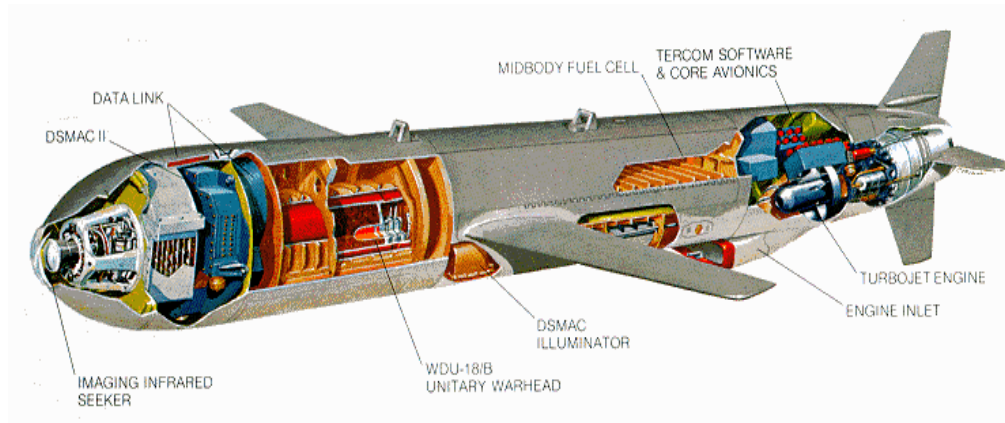


Figure 1. Tomahawk Land Attack Missile [Onwardoverland, 2004]

Some TLAM characteristics [Navy Office of Information, 2004.] are as follows:

- Unit Cost: Approximately \$569,000 (FY99 \$)
- Length: 18 feet 3 inches (5.56 m); with booster: 20 feet 6 inches (6.25 m)
- Weight: 2,900 pounds (1,315.44 kg); 3,500 pounds (1,587.6 kg) with booster
- Diameter: 20.4 inches (51.81 cm)
- Wing Span: 8 feet 9 inches (2.67 m)
- Speed: Subsonic - about 550 mph (880 km/h)
- Power Plant: Block II/III TLAM-A, C, D: Williams International F107 cruise Block IV TLAM-E: Williams International F415 cruise turbo-jet engine, ARC solid-fuel booster
- Range: Block II TLAM-A: 1,350 nm (1,500 statute miles, 2,500 km)
  - Block III TLAM-C: 900 nm (1,000 statute miles, 1,600 km)
  - Block III TLAM-D: 700 nm (800 statute miles, 1,250 km)
  - Block IV TLAM-E: 900 nm (1,000 statute miles, 1,600 km)
- Guidance System:
  - Block II TLAM-A: Inertial Navigation System (INS) Terrain Contour Matching (TERCOM)
  - Block III TLAM-C, D: INS TERCOM Digital Scene Matching Area Correlation Global Positioning Satellite System (GPS)
  - Block III TLAM-C, D: INS TERCOM GPS
- Warheads:
  - Block II TLAM-N: W80 nuclear warhead

- Block III TLAM-C: 1,000-pound class unitary warhead
- Block IV TLAM-E: 1,000-pound class unitary warhead
- Block III TLAM-D: Conventional submunitions dispenser with combined effect bomblets
- Date Deployed:
  - Block II TLAM-A: Initial Operational Capability (IOC) in 1984
  - Block III: IOC in 1994
  - Block IV: Expected 2004

## B. FIRING UNITS

TLAMs can be fired from ships and submarines. Some characteristics of these firing units are shown in Figure 2.

TLAM FIRING UNITS						
SHIPS	TYPE	Forward Launcher	Aft Launcher	Number of VLS Cells	Total Number of Half modules	Maximum Salvo Size
	Spruance Class Destroyer (DD-963)	Full	None	61	16	16
	Arleigh Burke Class Guided Missile Destroyer (DDG-51)	Half	Full	90	24	24
	Ticonderoga Class Guided Missile Cruiser (CG-47)	Full	Full	122	32	32
	Surface Combatant for the 21st Century (SC 21)	Planning to be replaced by Arleigh and Ticonderoga in 2004, with 128 VLS cells				
SUBMARINES (SSN)	TYPE	Installed CLS	Number of CLS Capsules	Number of Torpedo Tubes	Internal Weapon Capacity (at least)	Maximum Salvo Size
	LA 688-718	No	0	4	20	4
	LA 719-773	Yes	12	4	20	16
	Seawolf	No	0	8	40	8

Figure 2. TLAM Firing Units.

Ship Vertical Launching System (VLS) offer more storage capacity than submarines, which may or may not have a Canister Launching System (CLS). [Arnold, 2000]

### 1. Ships

Ships can store more TLAMs than submarines. They can also maintain better operational communication capabilities in order to coordinate TLAM missions with other

firing units and the Tomahawk Strike Coordinator (TSC). Ships can be considered as the main TLAM firing units. [Arnold, 2000.]

TLAMs on ships are launched from a Vertical Launching System (VLS). Depending on its type, a ship may have a forward launcher only or both aft and forward launchers. A launcher can be half-size or full-size depending on the type of ship. The basic component of a launcher is called a *module*. A module provides power and computer connections for each weapon to be fired from the cells in the module. Modules may be of standard type or crane type. The standard module is divided into two half-modules and each half-module is divided into four cells. In the crane module, the crane occupies cells “6” to “8” (see Figure 3). Thus, the crane module has a total of two half-modules and five cells. Each cell can store only one TLAM. Half-size launchers have four modules, eight half-modules, 29 cells and can store up to 29 TLAMs, whereas full-size launchers have eight modules, 16 half-modules, 61 cells and can store up to 61 TLAMs (see Figure 4). [Arnold, 2000.]

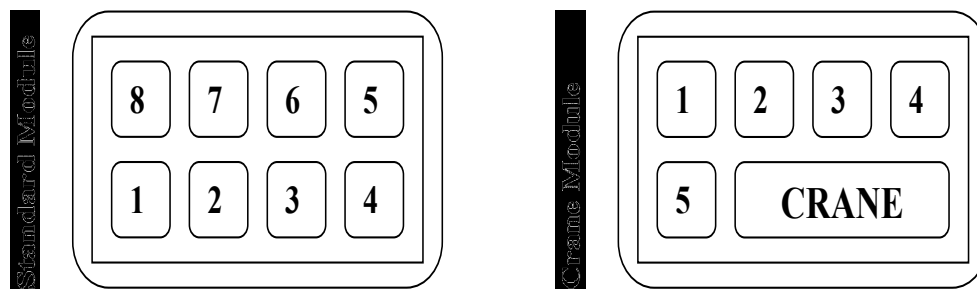


Figure 3. Standard and Crane Modules.

The basic unit of the VLS is a module. The standard module is divided into two half-modules consisting of cells “1-4” and “5-8” respectively. In a crane module, cells “6-8” are occupied by the crane and cell “5” is the only one available in the half-module, while the other half-module consists of cells “1-4.” [Arnold, 2000]

### Aft Launcher

### Forward Launcher

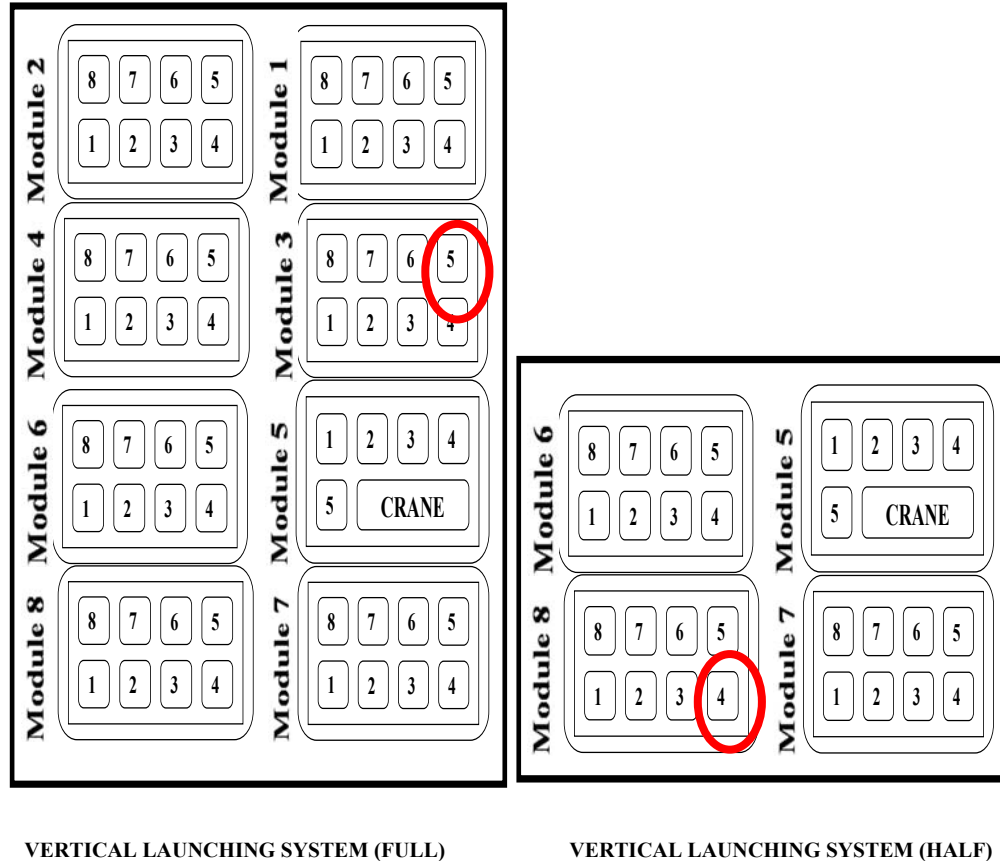


Figure 4. Full-Size and Half-Size VLS Layouts.

A full launcher consists of eight modules whereas a half-size launcher consists of four modules. Both launchers have one crane module with 5 cells. The circled cells are designated as “Aft, 3, 5,” and “Forward, 8, 4.” [Arnold, 2000]

Salvo size refers to the number of TLAMs that can be aligned and fired by a firing unit during any *conflict time window* (typically a 45-minute period). Due to VLS’s design and power limitations, only one missile can be fired from a half-module during a window. The tasks that are to be aligned within the same window are said to be *near-simultaneous* or to *conflict* with each other (see Section D for details). Because of half-module power constraints, a full-size launcher has a maximum salvo capability of 16 missiles during any window, while a half-size launcher’s salvo is eight missiles. [Arnold, 2000.]

## **2. Submarines**

Submarines can approach targets stealthily, but they must be close to the surface with antennas above the water when firing TLAMs. They reveal their location after the first firing. Submarines may have communication problems with the TSC in the operational area: for example, a TSC's short-notice order may not be received on time due to limited communication capability for security reasons. [Arnold, 2000.]

Submarines have two assets to align TLAMs: torpedo tubes and canister capsules. All U.S. Navy attack submarines (SSNs) have the capability of firing TLAMs from their torpedo tubes [See Figure 5]. A TLAM can be loaded into a torpedo tube, and then aligned and fired like a torpedo. Torpedo tubes can be reloaded with TLAMs carried on the submarine. All torpedo tubes on a submarine are capable of launching TLAMs. However, in order to maintain enough self-defense capability, a commanding officer might prefer not to dedicate all torpedo tubes to TLAM missions. [Arnold, 2000.]

In addition to torpedo tubes, Los Angeles class submarines with hull number of 719 (or higher) have a Capsule Launching System (CLS) that provides 12 canister capsules exclusively for firing TLAMs (see Figure 6) [Wingert, 2001]. The CLS on a submarine is similar to the surface ship's VLS and cannot be reloaded at sea. Unlike a ship VLS, a submarine CLS is not restricted by power constraints, so all missiles in CLS cells may be aligned, if necessary, within a conflict time window.



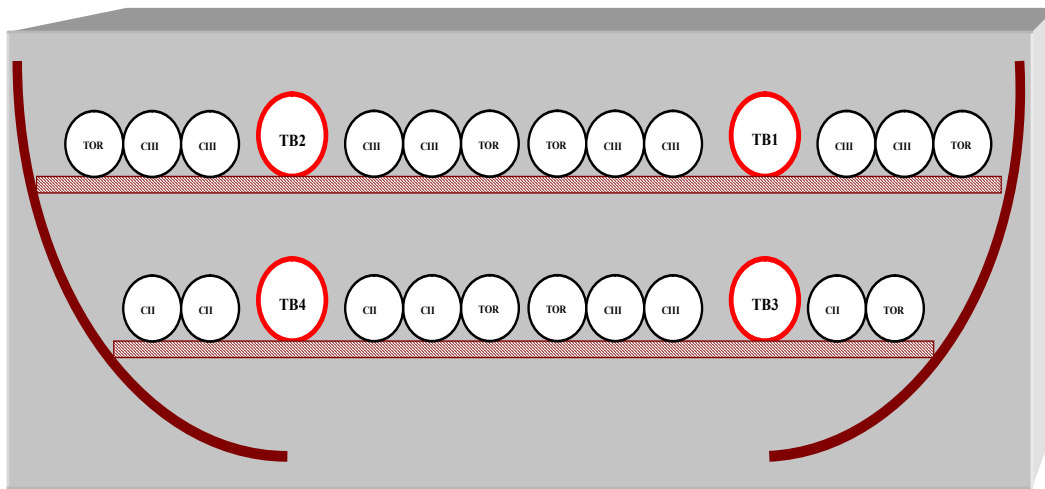


Figure 5. Cutaway View of a Torpedo Tube Room.

This Los Angeles Class Submarine has four torpedo tubes (TB1,...,TB4). The ordnance is stored in the torpedo room. Five CII TLAMs and ten CIII TLAMs ordnance are stored in this torpedo room. [Arnold, 2000]

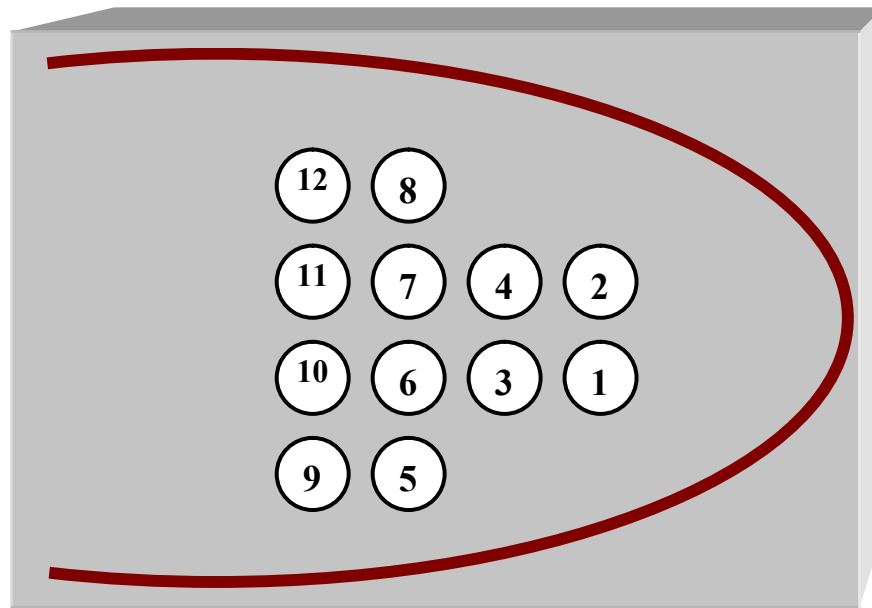


Figure 6. Capsule Launching System .

This CLS has 12 capsules; each can launch all current blocks and variants of TLAMs and is not limited by any power constraint. [Arnold, 2000]

### C. PREDESIGNATION

Predesignation is the assignment of land-attack missiles (on ships and submarines) to target aim-points. This decision process has two phases: (1) the allocation of the TLAMs to the firing units, considering all firing units and tasks simultaneously,

and (2) given the firing unit allocation from Phase 1, a specific allocation of tasks to missiles. The models presented in earlier works and in this thesis address the final allocation to specific missiles on firing units, which is the ultimate goal of Phase 2.

When the U.S. National Command Authority (NCA), formed by U.S. President and the Secretary of Defense, authorizes a strike, the authorization passes down through the chain of command via the Chief of Naval Operations (CNO) and a regional fleet Commander-in-Chief (CINC) and eventually to the Battle Group TSC. To begin Phase 1, the NCA or CINC provides the TSC the following information: a prioritized list of targets, a list of aim-points at the target, the desirable and feasible TLAM types, the number of missiles per aim-points, missile time on the target, and missile assignment redundancy requirement. [Arnold, 2000.]

The TSC predesignates the aim-points to firing units by selecting a set of pre-planned missions that meet NCA or CINC criteria. A pre-planned mission defines the target, aim-points, block, warhead and launch area [Arnold, 2000]. Predesignation considers the geo-graphical nearness of the candidate firing units to aim-points, the inventory and location of TLAMs aboard each firing unit, the limitations of preparing and firing particular missiles, the flight route coordination among TLAMs and other tactical concerns [Brown et al., 2001]. The other important issue to be considered by the TSC is to leave the combat units with a maximum residual salvo size after a strike, individually and as a battle group. The TSC tries to preserve as many remaining salvos as possible for each missile type on each firing unit that will remain with the battle group in the operation area. The TSC also avoids predesignations that interfere with other duties of the firing units and attempts to expend TLAMs from the firing units that will soon be departing the theater.

## **D. STRIKE PLANNING CONSIDERATIONS**

### **1. General Considerations**

A 10-character alphanumeric code, known as the Basic Encyclopedia Number (BEN), is used to designate each target chosen for attack. [Wingear, 2001.] For example, the BEN of “Target 1” may be 3004005000. The TSC determines the aim-points to refine the impact points of TLAMs on the target and the best TLAMs suited to

attack the targets. In addition, the TSC must determine the missions required to hit the aim-points he or she designates, the variants and total number of TLAMs to fire, and the desired execution time at each aim-point.

An 11-digit number known as the Mission Identification Number (MIN) specifies each task (also called mission) and associated list of missile variants that can be used to accomplish the assigned tasks. [Arnold, 2000.] Feasible missile variants for the task are listed on the Missile Mission Matching ( $M^3$ ) list. Every task has a list that indicates the relative desirability of all capable missile variants. The TSC should use the least capable missile in order to maximize residual salvo capabilities, as long as this choice does not interfere with other priorities. For instance, if the decision should be either to select a Block C-II or a Block C-III TLAM, a Block C-II is preferred since it has a less-capable warhead and shorter range than the Block C-III missile.

## **2. Near-Simultaneous Tasks**

As explained in Section B, the number of TLAMs fired from a VLS launcher on a ship is restricted by a power constraint, which allows only one TLAM to be aligned at once per half-module. From torpedo tubes on submarines this number is precisely the number of torpedo tubes. In this context, *at once* means within a pre-specified conflicting time window, typically 45 minutes.

A task has a launch time window defined by an earliest time to launch (ETL) and latest time to launch (LTL). [Kubu, 2001.] These are derived from the required arrival time at the aim-point and missile flight parameters. Although it would be desirable to fully employ this information to add flexibility to the predesignation process, our approach is a simplification of the time window specification. This is due to the fact that we will not optimize the exact time within the ETL-LTL window. Instead, we consider an average time to launch (ATL), calculated as the average of earliest and latest times to launch. If the average time to launch between any two tasks is less than the given conflicting time window, they are considered *near-simultaneous tasks*, and they are incorporated into a so-called *conflict task set* or *near-simultaneous task set*. A task may be in more than one of these sets.

The following example clarifies the concept of conflict sets and near-simultaneous tasks (see Figure 7). In this example, we assume the conflicting time window is 45 minutes. While some tasks are in only one near-simultaneous task set, others are in more than one of these sets, depending on their ATLs. Assuming that C-II and C-III are valid weapons for all of our tasks, by combining Figures 7 and 8 we conclude that, for example, the TSC cannot allocate tasks “150-150-15000” and “155-155-15500” to the module in Figure 8 due to power constraints. However, the TSC may allocate tasks “155-155-15500,” “160-160-16000” and “175-175-17500” to this module.

<b>TASK CODE (MIN)</b>	<b>ETL (Earliest Time to Launch)</b>	<b>LTL (Latest Time to Launch)</b>	<b>ATL (Average Time to Launch)</b>	<b>Average Time (min) Since 8/19/2000 00:00:00 AM</b>	<b>Conflict Task Set Number (s)</b>
150-150-15000	08/19/2000 00:00	08/19/2000 00:10	08/19/2000 00:05	5	1
155-155-15500	08/19/2000 00:00	08/19/2000 00:10	08/19/2000 00:05	5	1
160-160-16000	08/19/2000 00:30	08/19/2000 00:40	08/19/2000 00:35	35	1,2
165-165-16500	08/19/2000 00:30	08/19/2000 00:40	08/19/2000 00:35	35	1,2
170-170-17000	08/19/2000 00:30	08/19/2000 00:40	08/19/2000 00:35	35	1,2
175-175-17500	08/19/2000 01:00	08/19/2000 01:10	08/19/2000 01:05	65	2,3
180-180-18000	08/19/2000 01:00	08/19/2000 01:10	08/19/2000 01:05	65	2,3
185-185-18500	08/19/2000 01:00	08/19/2000 01:10	08/19/2000 01:05	65	2,3
190-190-17000	08/19/2000 01:00	08/19/2000 01:10	08/19/2000 01:05	65	2,3
225-225-22500	08/19/2000 01:30	08/19/2000 01:40	08/19/2000 01:35	95	3
230-230-23000	08/19/2000 01:30	08/19/2000 01:40	08/19/2000 01:35	95	3
235-235-23500	08/19/2000 01:30	08/19/2000 01:40	08/19/2000 01:35	95	3

Figure 7. Conflicting Sets of Tasks.

ATL is calculated by averaging ETL and LTL. We assume that two tasks whose ATLs are within 45 minutes of each other are near-simultaneous. In this example, tasks “150-150-15000” through “170-170-17000” are all conflicting with each other. Thus, they belong to conflict set #1. The same occurs for tasks “160-160-16000” through “190-190-17000,” which defines conflict set #2. Notice that some tasks belong to more than one conflict set [Excerpt from TLAM Documentation, 2002].

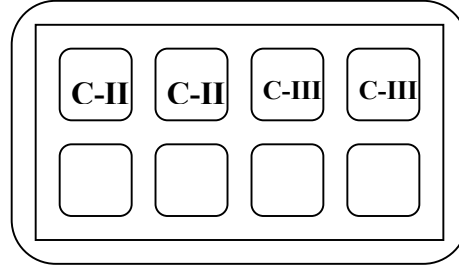


Figure 8. Module with an Empty Half-Module and Full Half-Module. The two C-IIs and two C-IIIs in the upper half-module can be used for non-near-simultaneous tasks. For example, only one of tasks “150-150-15000” and “155-155-15500” (from Figure 7) can be allocated to the upper half-module.

### 3. Missile Redundancy

One of the most important considerations for the TSC is to decide the amount of redundancy for each task of a target. Intuitively, the TSC wants to allocate more missiles to high priority targets in order to ensure their destruction in case some missiles fail to launch or miss the target. This leads to consideration of primary, ready-spare, and back-up task-parts, as well as a special type of task-parts called *ghost*. A primary task-part is designated to be launched at a target, whereas ready-spare and back-up task-parts are only executed in the case of primary task-part failure. Primary and ready-spare task-parts must be assigned to the same firing unit, whereas back-up task-parts must be assigned to a different firing unit than the primary and ready-spare (if any) task-parts. [Arnold, 2000.]

Multiple missiles may be required for each task-part. The ready-spare and back-up task-parts are aligned simultaneously with the primary task-part. Ready-spare task-parts are launched immediately if the missile associated with the primary task-part fails, and back-up task-parts are launched if the missiles associated with the primary task-part and ready-spare (if any) fail. Ghost task-parts can only be assigned to a special type of tasks, called *ghost tasks*. A ghost task enables one missile to provide redundancy for multiple tasks and multiple task-parts concurrently. A ghost task is fired when ordered by the TSC. For some calculations (such as expected residual salvo), we assume that all missiles associated with primary task-parts are fired, and those associated with ready-spare, back-up and ghost task-parts are not. [Arnold, 2000.]

#### 4. Other Considerations

There are some doctrinal constraints applying to targets and tasks. Each target is designated either as restricted or unrestricted. All the task-parts in tasks associated with unrestricted targets may be spread among the geo-feasible firing units. Restricted targets have their tasks and associated task-parts allocated to a limited number of firing units, which is specified by the TSC. In addition, if any task-part of a task in a restricted target is left unassigned, all the other task-parts of that task will also be considered as unassigned. [Arnold, 2000.]

For each task-part with multiple missile requirements the TSC specifies whether the missiles for the task-part are separable (formerly referred to as *split*) or not across firing units. If missiles of any task-part are non-separable, all missiles of that task-part should be assigned to the same firing unit. For example, in Figure 9, “Target A” has two tasks: “Task 1” and “Task 2.” These tasks have their own task-parts (e.g., there are two primary missiles required for “Task 1”).

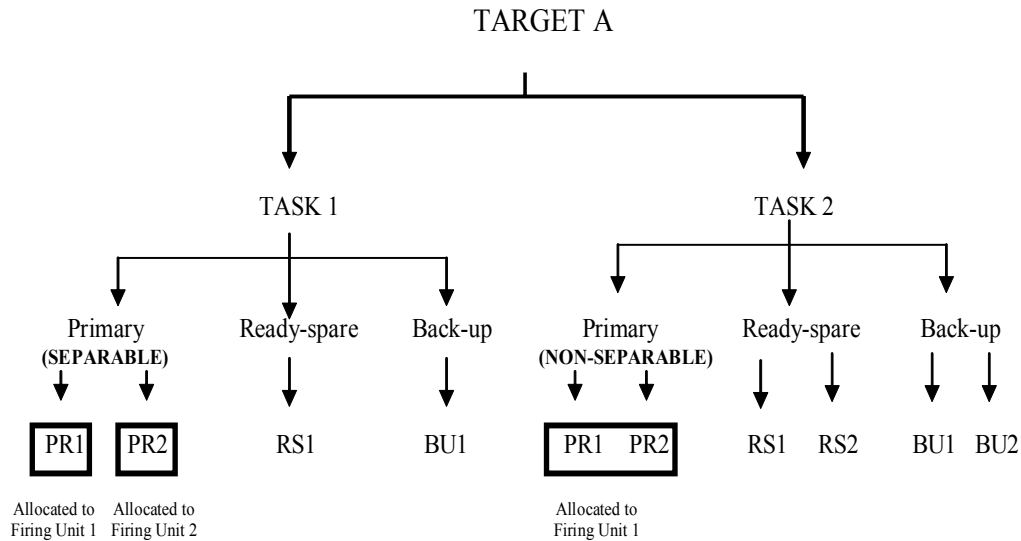


Figure 9. Separable and Non-separable Tasking.

In the figure, primary task-parts in “Task 1” are deemed “separable,” but primary task-parts in “Task 2” are “non-separable.” Thus, “PR1” and “PR2” in “Task 1” can be allocated either to the same firing unit or to different firing units, whereas “PR1” and “PR2” in “Task 2” must be allocated to the same firing unit.

If the primary task-part in “Task 1” is deemed *separable*, we can allocate both “PR1” and “PR2” to either the same firing unit or to different firing units. On the other hand, if the primary task-part in “Task 2” is deemed non-separable, both “PR1” and “PR2” must be allocated to the same firing unit.

## **E. SUMMARY OF PAST WORK**

Since 1998, a group of faculty and students at the Naval Postgraduate School (NPS) Operations Research Department has contributed to the development of a decision support algorithm for Tomahawk Predesignation, under the sponsorship of the Naval Surface Warfare Center, Dahlgren Division (NSWCDD). A series of student theses documents this effort. Most of these theses were advised by Professor Alexandra M. Newman, now with the Colorado School of Mines.

LT Scott Kuykendall [1998] breaks ground by addressing Tomahawk predesignation with a mixed-integer programming (MIP) formulation with elegant simplicity. LT Kuykendall’s fleet predesignation experience motivated him as follows: (i) current paper-and-pencil predesignation process is inefficient and error-prone, and (ii) current predesignation software is neither user-friendly nor satisfactory, since it produces predesignations that can be trivially improved by inspection [Wu et al., 2002]. Kuykendall’s work considers a single-ship scenario and aims to maximize residual strike capability (also referred to as *follow-on salvo* or *residual salvo*). He proves this scheme yields better solutions than the manual predesignation method; yet, his MIP does not address all relevant objectives.

LT Brian Kirk [1999] develops a multi-objective (MO) MIP for automatic surface ship TLAM Phase I predesignation at the fleet level. His model is based on eight objectives, reviewed by Tactical Training Groups Atlantic and Pacific, presented below in descending order of importance [Wu et al., 2002]:

- Minimize the number of unassigned tasks (actually task-parts).
- Minimize the use of units already occupied with other operations or not in geo-graphical proximity to the strike.
- Maximize the allocation of tasks across so-called *expend* firing units.
- Use the missiles from firing units designated as non-expend so as to level the number of missiles remaining across these firing units.

- Spread primary task-part missiles across as many firing units as possible to prevent single-point failures among primary task-part allocation.
- Spread back-up task-part missiles across as many firing units as possible to prevent single-point failures among back-up task-part allocation.
- Use the least capable missile possible for each mission.
- Maximize the residual salvo capability.

Kirk applies three solution methods. First, he attempts to solve the model as a single, monolithic problem consisting of a combination all the objectives into a single objective function, where higher-priority goals have greater weights. He finds that this technique does not produce sensible answers for large scenarios, and solution times are prohibitive. Next, he implements a Hierarchical Restriction by solving for a goal at a time. He achieves solutions of reasonable quality by using this technique but solution times are still too large to be operationally useful. In order to lessen the solution times, Kirk implements a Heuristic Hierarchical Restriction method on the MO MIP, which produces faster solutions compared to those of previous techniques, but not fast enough to be used in an operational setting. [Wu et al., 2002.]

LT Bertram Hodge [1999] introduces a new heuristic, based on a prioritized target list that he uses to mimic the optimal decisions of Kirk's most comprehensive model. [Brown et al., 2001.] This fast, heuristic algorithm selects firing units and then predesignates targets from the list in a single pass. Hodge uses an algorithm to select ships and then performs task allocations based on a single pass of a prioritized task list. His heuristic produces solutions roughly 20% worse than Kirk's Hierarchical Restriction for all objective function values. As a result, his model reduces solution times at the cost of solution quality.

LT Andrew Arnold [2000] improves the performance of Hodge's heuristic by modifying the heuristic algorithm to improve solution quality. Arnold's heuristic extends Hodge's fast heuristic to include submarines as firing units and ghost missions as task-parts. Arnold introduces the concepts of Target (as a group of tasks with common constraints), as well as the possibility of multiple missiles assigned for any task-part. He also introduces the concept of missile separability by task-parts, and allows the TSC to specify part of the allocation. [Wu et al., 2002]



LT Justin A. Kubu [2001] enhances Arnold's heuristic by introducing diagnoser and prescriber modules that provide the TSC with possible reasons why a task cannot be allocated by the heuristic algorithm. Although the heuristic cannot determine whether a feasible allocation under the given conditions exists, it prescribes modifications to facilitate a complete allocation of tasks. Prescriptions include, for example, shifting the launch time of a task for situations in which the heuristic predesignation does not provide a complete allocation.

LT Paul H. Wingert [2001] enhances Arnold's heuristic by implementing two computer-assisted allocation methods: (i) task-part to firing unit, and (ii) firing unit to task-part. Wingert validates his heuristic solutions with exercise data TEXAS THUNDER 00-6 and RIMPAC 00-3 and shows that automated allocation yields solutions superior to those of manual predesignation in these fleet exercises.

Consolidated specifications for the problem data and heuristic algorithm can be found in TLAM documentation [2002].

## **F. THESIS OUTLINE**

The rest of the thesis is organized as follows: Chapter II introduces the problem specifications that serve as basis for our mathematical formulation. Then we introduce the problem formulation as a MIP MO optimization model. We present the so-called hierarchical and combined-goal formulations for the problem. Chapter III summarizes the results for all test cases from our implementation in GAMS [Brooke et al., 1996]. We also describe a variable partition technique called *Fix-and-Relax* to accelerate convergence in some cases. Chapter IV presents our conclusions and recommendations for future work.

THIS PAGE INTENTIONALLY LEFT BLANK

## II. MIXED-INTEGER LINEAR OPTIMIZATION MODEL

In this chapter, we present the mathematical formulation of our TLAM predesignation problem as a MO MIP optimization model. To set the stage for the model formulation, we first list the complete specifications of the predesignation problem, as addressed in the remainder of the thesis.

### A. PROBLEM SPECIFICATIONS: DEFINITIONS AND ASSUMPTIONS

#### 1. Near-Simultaneous Tasks

- Each *task* has an *earliest time* and a *latest time* to launch (ETL, LTL).
- An *average time to launch* (ATL) (calculated as the midpoint between ETL and LTL) can be used as hypothetical time to determine *near-simultaneous* (also called *conflicting*) tasks.
- If the difference in the ATL for two given tasks is less than the given *conflicting time window* (typically 45 minutes), both tasks must belong to a common *near-simultaneous task set*.
- A task may be in more than one near-simultaneous task set.

#### 2. Firing Units

- There are two types of *firing units*: *ships* and *submarines*.
- Some firing units incur a so-called *employment penalty*. (Related objective: Goal #2.)
- Each firing unit is defined as either *expend* or *non-expend*. It is preferable to use weapons from expend firing units. (Related objective: Goal #3.)

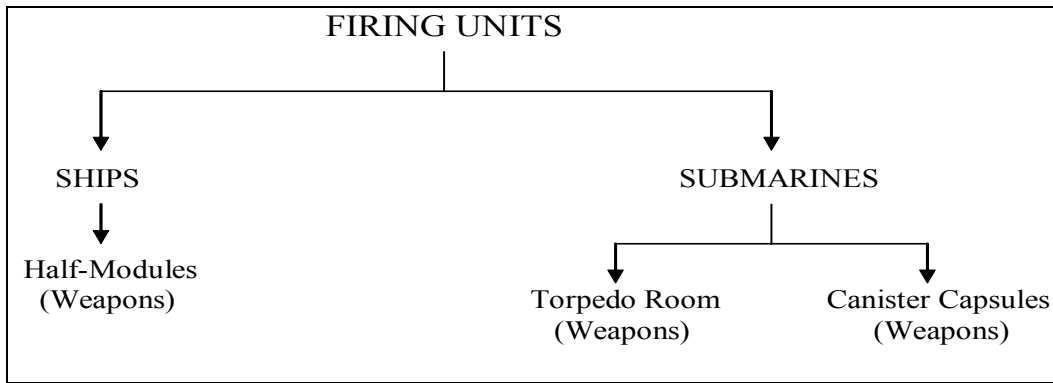


Figure 10. Firing Units.

There are two types of firing units: Ships and submarines. Ships stores weapons in cells in half-modules, whereas submarines have two types of storage for weapons: torpedo room and canister capsules.

### **3. Ships**

- Each ship has a given number of *half-modules*, depending on ship type as shown in Figure 2. Each half-module may contain up to four TLAMs, and these can be of different type (see weapon types below).
- The number of TLAMs fired from a ship is restricted by a power constraint, which allows only one TLAM to be aligned per half-module within a conflicting time window.

### **4. Submarines**

- Each submarine can fire TLAMs from its torpedo tubes and from its canister capsules (if any). The number of tubes and capsules depends on the submarine type.
- Torpedo tubes can fire the TLAMs available in the torpedo room. The number and type of weapons in the torpedo room are submarine dependent.
- The maximum number of weapons that can be fired from the torpedo tubes per conflicting time window is precisely the number of torpedo tubes.
- Each canister capsule can fire only one weapon (the one that is already loaded in the capsule, if any).
- Canister capsules on a submarine are not affected by conflicting time windows.
- Each submarine may or may not be designated for *back-up task-part* allocation by the Tomahawk Strike Coordinator (TSC).

### **5. Targets**

- Each *target* consists of one or more *tasks*.
- Each target is designated either as *restricted* or *unrestricted*. A restricted target needs all tasks and *task-parts* to be allocated to a limited number of firing units, given independently for each restricted target. An unrestricted target has no limitation in this regard.
- A target is deemed allocated when all of its tasks have been allocated. If the target is unrestricted, it is allowed (although not desirable) that some tasks in the target are allocated while others are not. However, if the target is restricted, then all or none tasks of the target tasks should be allocated.

### **6. Tasks, Task-Parts and Task-Part Missiles**

- Each task belongs to a specific target.
- There are four types of task-parts: primary (PR), ready-spare (RS), back-up (BU) and ghost (GH). (See List of Terms and Abbreviations.)
- Each task may have one or more types of task-parts.
- Each task-part in a task requires one or more missiles.

- Every task must have either a PR task-part, or a GH task-part, but not both. The former is called a regular task, and may also include the following types of task-parts: RS and/or BU. If the task has a GH task-part, it is called a *ghost task*, and it cannot include any other type of task-part (see Figure 11).
- For regular tasks, there cannot be more RS missiles required than PR missiles required.
- A task is considered allocated only when all of its task-parts have been fully allocated, i.e., when all the missiles required for each of its task-parts have been allocated. Otherwise, the task is deemed unallocated. (Partial allocation of a task is not allowed.) (Related objective: Goal #1.)
- PR and RS task-parts must be allocated to the same firing unit in pairs: PR1-RS1 first; then PR2-RS2, and so forth. The  $n$ th primary task-part ( $PR_n$ ) can be allocated alone if  $RS_n$  is not required.
- There is no limit in the number of missiles required for a BU task-part in a regular task, nor it is required that a RS task-part exists for the task in order to require one or several BU task-parts.
- BU task-parts (for a given task) must be allocated to firing units excluding all of the firing units used for the PR task-parts (for the same task).
- Each task-part of each task is classified as *separable* or *non-separable*. All the missile requirements of a non-separable task-part must be allocated to the same firing unit. (Separable and Non-separable are sometimes referred to as *split* and *non-split*, respectively)

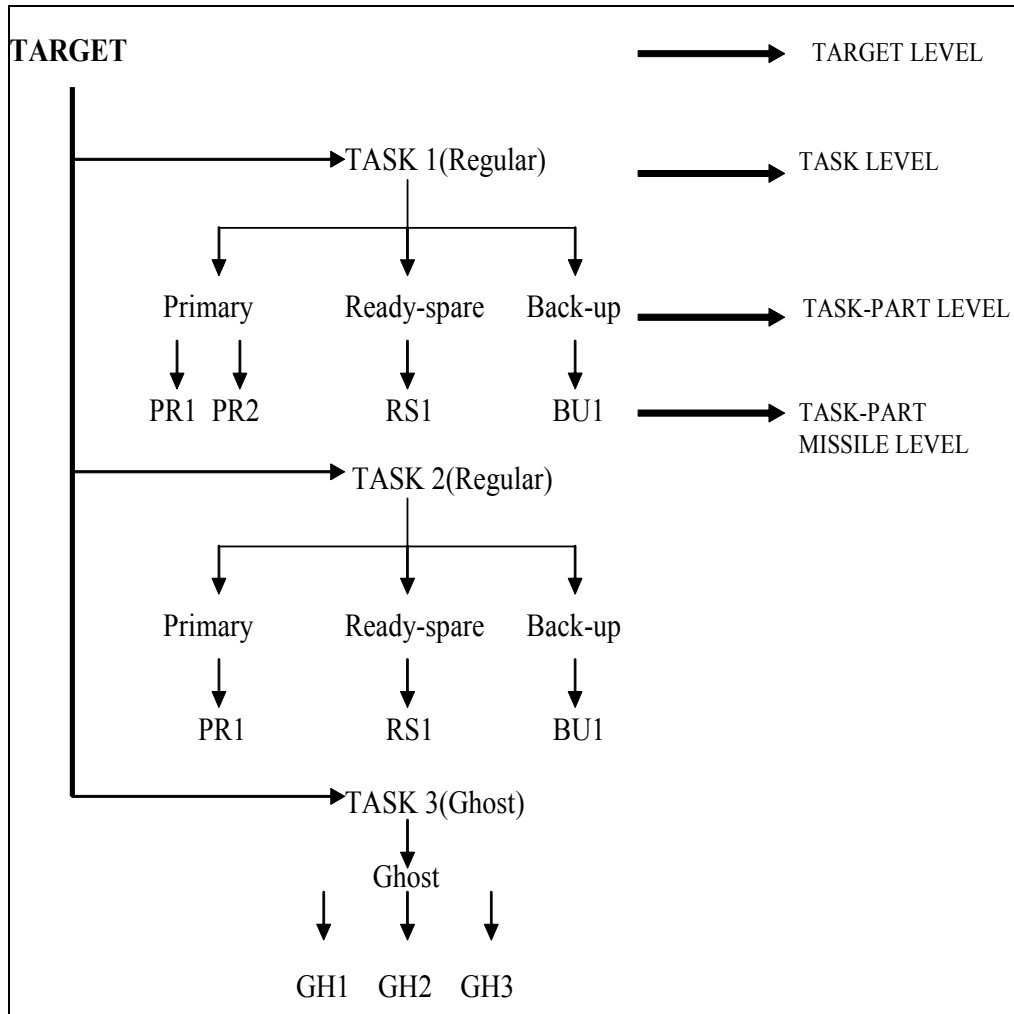


Figure 11. Tasking Levels.

## 7. Launch Areas

- Each firing unit is assumed to be located in a unique *launch area*, and cannot be moved to a different area.
- Each launch area has a list of targets that can be attacked from the area. A firing unit in the launch area of a target is considered *geo-feasible*.
- Primary task-parts allocated to firing units in a particular area can be characterized as *spread* or *not spread*. This refers to whether or not spreading the primary task-parts across the different firing units in the area is desired. (Related objective: Goal #5.)
- Back-up task-parts allocated to firing units in a particular area can also be characterized as *spread* or *not spread*. (Related objective: Goal #6.)

## **8. Weapons**

- Each missile required by any task-part in a given task must be allocated a weapon type from the so-called  $M^3$  (missile to mission matching) list. This list is task-dependent.
- Each available weapon in the  $M^3$  list for the task has an associated priority to be used that makes it more or less desirable to use compared to the other weapons on the list. (Related objective: Goal #7.)
- Each weapon type has a value that is used to compute the residual salvo value after allocation. Residual salvo computation must take into account half-module and torpedo tube power-constraints. Thus, only the most valuable weapon from each half-module, and a limited number of weapons from the torpedo room (as many as torpedo tubes) must count for the residual salvo computation. (Related objective: Goal #8.)

## **9. Tomahawk Strike Coordinator Allocation**

The TSC may specify mandatory allocation of selected task-part missile requirements by providing, for each of these requirements, the following information:

- A ship, a half-module, a weapon type (for allocation to ships), or
- A submarine and either a weapon type in the torpedo room or a weapon type in canister capsules (for allocation to torpedo tubes or canister capsules on submarines, respectively).

## **10. Goal Specifications**

### ***a. Goal #1***

“Meet All Assigned Tasking” - To date, this goal has been addressed by minimizing the number of “unassigned task-part missile requirements.” This interpretation is assumed in our formulation.

In some cases, however, it might be desirable to address Goal #1 by minimizing some of the following:

- A weighed average of unmet tasks-parts. For example, consider the following task-part missile requirements:
  - Task 1: PR1, BU1, BU2, BU3
  - Task 2: PR1, PR2, PR3

Since a task cannot be allocated unless all of its task-parts are allocated, currently it is deemed more important to allocate the four missiles required for Task 1

(even if three of them are BUs) than the three missiles required for Task 2, all of which are PRs. This can be remedied by weighing the task-parts.

- The number of unmet tasks, instead of the number of unmet task-parts. These two outputs might be significantly different, depending on how evenly the missile requirements are spread among the tasks.
- The number of targets that cannot be fully allocated. (*Non-restricted* targets allow some tasks to be allocated even if other tasks in the target cannot be allocated.)

**b. Goal #2**

“Minimize the Use of Firing Units Currently Engaged in Other Tasking” -  
Currently, this goal has been accomplished with the use of *employment penalties*.

**c. Goal #3**

“Maximize Missiles Allocated from Designated Expend Firing Units” -  
Remark: Only allocated missiles for PR task-parts are deemed used. That is, even if other RS, BU and GH missiles are allocated, these do not become part of the preliminary count of fired missiles.

**d. Goal #4**

“Level the Number of Missiles Remaining on Non-Expend Firing Units” -  
Remarks: Unallocated missiles along with those missiles allocated to RS, BU and GH task-parts are deemed remaining. The *level* definition is based on the total difference between the arithmetic average of the remaining missiles across all non-expend firing units and the remaining missiles on each non-expend firing unit (even if it is a firing unit with positive employment penalty).

**e. Goal #5**

“Spread Primary Task-Part Missiles Among As Many Firing Units As Possible, in Launch Areas So Designated.”

**f. Goal #6**

“Spread back-up task-part missiles among as many firing units as possible, in launch areas so designated.”

**g. Goal #7**

“From among all feasible missile types for each task, choose the least capable missile, so as to preserve more capable missiles for the future.”



***h. Goal #8***

“Maximize residual salvo capability” - Remark: Only allocated PR task-parts are used to compute the post-allocation salvo value. To do this for ships, we use the most valuable weapon in each half-module. For submarines, we use the most valuable weapons in the torpedo room that can be fired simultaneously from the torpedo tubes, and all the weapons in the canister capsules.

**B. TLAM ALLOCATION MODEL**

In this section, we present the problem formulation. In order to simplify the presentation, we have divided the formulation into several parts driven by the goal and/or the feature in the specifications: Initially, we present a basic formulation for the first goal with limited problem features. Then, we add other goals and specifications such as submarines, restricted targets, TSC specifications, etc. The full Allocation Model is shown in Appendix A.

**1. Basic Model with Goal #1**

In this model, we only consider ships as firing units, and goal #1 in the objective function.

• Indices

$S$ ,	set of ships; $s \in S$
$H$ ,	set of half-modules; $h \in H = \{1, 2, 3, \dots\}$
$G$ ,	set of targets; $g \in G$
$T$ ,	set of tasks; $t \in T$
$P$ ,	set of task-parts; $p \in P = \{PR, RS, BU, GH\}$
$M$ ,	set of task-part missile requirements; $m, m' \in M = \{1, 2, 3, \dots\}$
$W$ ,	set of weapon types; $w \in W$
$I$ ,	set of subsets of near-simultaneous tasks; $i \in I$
$A$ ,	set of launch areas; $a \in A$
$K$ ,	set of goals; $k \in K = \{1, 2, \dots, 8\}$ . (Remark: We introduce the set of all goals although we only formulate goal #1 here. The rest of the goals will be formulated later.)

- Sets

$T_g^G$ , subset of tasks for target  $g$ ;  $T_g^G \subset T$

$T_i^I$ , subset of near-simultaneous tasks in group  $i$ ;  $T_i^I \subset T$

$G_a^A$ , subset of targets which are geo-feasible from area  $a$ ;  $G_a^A \subset G$

$W_t^T$ , subset of weapons that can be used for task  $t$ ;  $W_t^T \subset W$

- Data

$n_{shw}^S$ , number of weapons of type  $w$  in half-module  $h$  on ship  $s$

$m_s^H$ , number of half-modules on ship  $s$

$n_{tp}^M$ , number of missiles required for task-part  $p$  of task  $t$

$a_s^S$ , area for ship  $s$

- Derived Data and Sets

$g_t$ , target for task  $t$  (i.e.,  $t \in T_g^G$ )

$S_t^T$ , subset of geo-feasible ships for task  $t$ ;  $S_t^T = \left\{ s \mid g_t \in G_{a_s^S}^A \right\}$

- Decision Variables

$X_{shwtpm}^S$ , equals 1 if a weapon of type  $w$  in half-module  $h$  of ship  $s$  is used to allocate missile  $m$  of task-part  $p$  in task  $t$ , 0 otherwise

$Y_{stpm}^S$ , equals 1 if missile  $m$  of task-part  $p$  in task  $t$  is allocated to ship  $s$ , 0 otherwise

$U_{tpm}$ , equals 1 if task  $t$  is unmet, 0 otherwise

$Z_k$ , objective function value for goal  $k \in K$ . (Remark: We introduce decision variables for all objective functions here although we only use goal #1 in this model).

- Formulation

Goal  $k = 1$ : Minimize the total number of unmet task-parts:

$$Z_1^* = \min Z_1, \text{ where } Z_1 = \sum_t \sum_p \sum_{m=1}^{n_{tp}^M} U_{tpm}$$

Subject to:

- Ensure weapon availability for tasks allocated to half-modules:

$$1.1. \sum_{t \left| \begin{smallmatrix} s \in S_t^T, \\ w \in W_t^T \end{smallmatrix} \right.} \sum_p \sum_{m=1}^{n_{tp}^M} X_{shwtpm}^S \leq n_{shw}^S, \quad \forall s, h, w \mid 1 \leq h \leq m_s^H, n_{shw}^S > 0$$

- Establish the allocation at the firing unit level:

$$1.2. \sum_{h=1}^{m_s^H} \sum_{w \in W_t^T \mid n_{shw}^S > 0} X_{shwtpm}^S = Y_{stpm}^S, \quad \forall s, t, p, m \mid s \in S_t^T, 1 \leq m \leq n_{tp}^M$$

$$1.3. \sum_{s \in S_t^T} Y_{stpm}^S \leq 1, \quad \forall t, p, m \mid 1 \leq m \leq n_{tp}^M$$

- Ensure all (or none) of the task-parts in a task are allocated:

$$1.4. \sum_{s \in S_t^T} Y_{stp'm'}^S = \sum_{s \in S_t^T} Y_{stpm}^S, \quad \forall t, p, m \mid 1 \leq m \leq n_{tp}^M, n_{tp'}^M > 0, \\ \text{for } p' \in \{ "PR", "GH" \}, m' = "1"$$

- Prevent near-simultaneous tasks in the same half-module:

$$1.5. \sum_{w \mid n_{shw}^S > 0} \sum_{t \in T_i^T \mid \left| \begin{smallmatrix} s \in S_t^T, \\ w \in W_t^T \end{smallmatrix} \right.} \sum_p \sum_{m=1}^{n_{tp}^M} X_{shwtpm}^S \leq 1, \quad \forall s, h, i \mid 1 \leq h \leq m_s^H$$

- Allocate each task-part missile requirement or take a penalty:

$$1.6. \sum_{s \in S_t^T} Y_{stpm}^S + U_{tpm} = 1, \quad \forall t, p, m \mid 1 \leq m \leq n_{tp}^M$$

- Ensure primary and back-up task-part relations:

$$1.7. Y_{stpm}^S + Y_{stp'm'}^S \leq 1, \quad \forall s, t, m, m' \mid s \in S_t^T, 1 \leq m \leq n_{tp}^M, 1 \leq m' \leq n_{tp'}^M, \\ \text{for } p = "PR", p' = "BU"$$

- Ensure primary and ready-spare task-part relations:

$$1.8. Y_{stpm}^S = Y_{stp'm}^S, \quad \forall s, t, m \mid s \in S_t^T, 1 \leq m \leq n_{tp}^M, \\ \text{for } p = "PR", p' = "RS"$$

- Binary variables:

$$1.9. \quad X_{shwtpm}^s \in \{0,1\}, \quad \forall s, h, w, t, p, m \mid s \in S_t^T, 1 \leq h \leq m_t^H, w \in W_t^T, t \in T, n_{shw}^S > 0, p \in P, 1 \leq m \leq n_p^M$$

- Non-negative variables and bounds:

$$1.10. \quad 0 \leq Y_{stpm}^S \leq 1, \quad \forall s, t, p, m \mid s \in S_t^T, t \in T, p \in P, 1 \leq m \leq n_p^M$$

$$1.11. \quad 0 \leq U_{tpm} \leq 1, \quad \forall t, p, m \mid t \in T, p \in P, 1 \leq m \leq n_p^M$$

## 2. Goal #2

- Data

$p_s^S$ , employment penalty for ship  $s$

- Decision Variables

$ZZ_{sg}^S$ , equals 1 if ship  $s$  is engaged in any tasking in target  $g$ , 0 otherwise

$Z_s^S$ , equals 1 if ship  $s$  is used, 0 otherwise

- Formulation

Goal  $k = 2$ : Minimize the use of *penalty* firing units:

$$Z_2^* = \min Z_2, \text{ where } Z_2 = \sum_s p_s^S Z_s^S$$

Additional constraints:

- Calculate firing unit engagement in targets and in the overall allocation:

$$2.1. \quad Y_{stpm}^S \leq ZZ_{sg}^S, \quad \forall s, t, p, m, g \mid s \in S_t^T, 1 \leq m \leq n_p^M$$

$$2.2. \quad ZZ_{sg}^S \leq Z_s^S, \quad \forall s, g$$

(Remark: Engagement is calculated at the level of targets for its use in other specifications later in the formulation. For the purpose of calculating firing unit engagement in the allocation only, (2.1) and (2.2) can be combined into:

$$Y_{stpm}^S \leq Z_s^S, \quad \forall s, t, p, m \mid s \in S_t^T, 1 \leq m \leq n_p^M)$$

- Binary variables:

$$2.3. \quad ZZ_{sg}^S \in \{0,1\}, \quad \forall s, g \mid s \in S, g \in G$$

$$2.4. Z_s^S \in \{0,1\}, \quad \forall s \in S$$

### 3. Goal #3

- Sets

$S^{Exp}$ , subset of expend ships;  $S^{Exp} \subset S$

- Decision Variables

$E^S$ , total number of weapons allocated to primary task-parts from *expend* ships

- Formulation

Goal  $k = 3$ : Maximize the number of weapons allocated from designated *expend* firing units:

$$Z_3^* = \max Z_3, \text{ where } Z_3 = E^S$$

Additional constraints:

- Use of expend firing units for primary task-parts:

$$3.1. \sum_{s \in S^{Exp}} \sum_{t|s \in S_t^T} \sum_{m=1}^{n_{tp}^M} Y_{stpm}^S = E^S, \quad \text{for } p = "PR"$$

- Non-negative variables:

$$3.2. E^S \geq 0$$

### 4. Goal #4

- Data

$n^{NExp}$ , number of non-expend firing units:  $n^{NExp} = |S \setminus S^{Exp}|$

- Decision Variables

$RN_s^S$ , number of missiles on *non-expend* ship  $s$  that are not allocated to a primary task-part

$RAVG$ , average of the residual number of missiles across non-expend firing units after allocation

$RD_s^S$ , difference between the residual number of missiles on ship  $s$  and  $RAVG$

- Formulation

Goal  $k = 4$ : Minimize the deviation from average of residual number of missiles on non-expend firing units:

$$Z_4^* = \min Z_4, \text{ where } Z_4 = \sum_{s \in S \setminus S^{Exp}} DR_s^S$$

Additional constraints:

- Compute absolute deviation from average number of residual weapons onboard each firing unit:

$$4.1. \quad \sum_{h=1}^{m_s^H} \sum_w n_{shw}^S - \sum_{t|s \in S_t^T} \sum_{m=1}^{n_p^M} Y_{stpm}^S = RN_s^S, \quad \forall s \in S \setminus S^{Exp}, \quad \text{for } p = "PR"$$

$$4.2. \quad \sum_{s \in S \setminus S^{Exp}} RN_s^S = n^{NExp} RAVG$$

$$4.3. \quad RN_s^S - RAVG \geq -RD_s^S, \quad \forall s \in S \setminus S^{Exp}$$

$$4.4. \quad RN_s^S - RAVG \leq RD_s^S, \quad \forall s \in S \setminus S^{Exp}$$

- Non-negative variables:

$$4.5. \quad RN_s^S \geq 0, \quad \forall s \in S$$

$$4.6. \quad RD_s^S \geq 0, \quad \forall s \in S$$

$$4.7. \quad RAVG \geq 0$$

## 5. Goal #5

- Sets

$A^{PR}$ , subset of areas whose primary task-parts are desired to be assigned to as many firing units as possible;  $A^{PR} \subset A$

- Decision Variables

$PR_s^S$ , equals 1 if ship  $s$  is assigned to allocated any primary task-part, 0 otherwise

- Formulation

Goal  $k = 5$ : Maximize the number of firing units that have been assigned to allocate primary task-parts in launch areas so designated:

$$Z_5^* = \max Z_5, \text{ where } Z_5 = \sum_{a \in PR^A} \left( \sum_{s|a_s^S=a} PR_s^S \right)$$

Additional constraints:

- Keep track of areas that must spread primary task-part missiles:

$$5.1. \sum_{t|s \in S_t^T} \sum_{m=1}^{n_p^M} Y_{stpm}^S \geq PR_s^S, \quad \forall s | a_s^S \in A^{PR}, \quad \text{for } p = "PR"$$

- Binary variables:

$$5.2. PR_s^S \in \{0,1\}, \quad \forall s \in S$$

## 6. Goal #6

- Sets

$A^{BU}$ , subset of areas whose back-up task-parts are desired to be assigned to as many firing units as possible;  $A^{BU} \subset A$

- Decision Variables

$BU_s^S$ , equals 1 if ship  $s$  is assigned to any back-up task-part, 0 otherwise

- Formulation

Goal  $k = 6$ : Maximize the number of firing units that have been assigned to allocate back-up task-parts in launch areas so designated:

$$Z_6^* = \max Z_6, \text{ where } Z_6 = \sum_{a \in BU^A} \left( \sum_{s|a_s^S=a} BU_s^S \right)$$

Additional constraints:

- Keep track of areas that must spread primary task-part missiles:

$$6.1. \sum_{t|s \in S_t^T} \sum_{m=1}^{n_p^M} Y_{stpm}^S \geq BU_s^S, \quad \forall s | a_s^S \in A^{BU}, \quad \text{for } p = "BU"$$

- Binary variables:

$$6.2. BU_s^S \in \{0,1\}, \quad \forall s \in S$$

## 7. Goal #7

- Data

$m_p^3$ , weight of task-part  $p$  to match the  $M^3$  list position of for task-part  $p$

$m_{tw}^3$ ,  $M^3$  position of weapon  $w \in W_t^T$  for task  $t$

- Formulation

Goal  $k = 7$ : Minimize the *missile to mission matching* ( $M^3$ ) list position of all selected weapons:

$$Z_7^* = \min Z_7, \text{ where } Z_7 = \sum_s \sum_{h=1}^{m_s^H} \sum_{w|n_{shw}^S > 0} \sum_{\substack{t|s \in S_t^T, \\ w \in W_t^T}} \sum_p \sum_{m=1}^{n_{tp}^M} m_p^3 m_{tw}^3 X_{shwtpm}^S$$

## 8. Goal #8

- Data

$v_w$ , relative value of weapon  $w$  for residual salvo computation

- Decision Variables

$D_{shw}^S$ , equals 1 if one or more weapons of type  $w$  remain on ship  $s$  in half-module  $h$  after firing all primary task-part missiles, 0 otherwise

- Formulation

Goal  $k = 8$ : Maximize residual salvo capability:

$$Z_8^* = \max Z_8, \text{ where } Z_8 = \sum_s \sum_{h=1}^{m_s^H} \sum_{w|n_{shw}^S > 0} v_w D_{shw}^S$$

Additional constraints:

- Calculate residual salvo size:

$$8.1. n_{shw}^S - \sum_{\substack{t|s \in S_t^T, \\ w \in W_t^T}} \sum_{m=1}^{n_{tp}^M} X_{shwtpm}^S \geq D_{shw}^S, \quad \forall s, h, w | 1 \leq h \leq m_s^H, n_{shw}^S > 0, \text{ for } p = "PR"$$

$$8.2. \sum_{w|n_{shw}^S > 0} D_{shw}^S \leq 1, \quad \forall s, h | 1 \leq h \leq m_s^H$$

- Non-negative variables and bounds:

$$8.3. 0 \leq D_{shw}^S \leq 1, \quad \forall s, h, w | s \in S, 1 \leq h \leq m_s^H, w \in W_t^T, n_{shw}^S > 0$$



## 9. Restricted Targets

- Sets

$G^R$ , subset of targets which are designated as *restricted*;  $G^R \subset G$

- Data

$n_g^{FU}$ , maximum number of firing units to which a restricted target  $g$  can be allocated

- Formulation

Additional constraints:

- Calculate firing unit engagement in the target:

Same as constraint (2.1).

$$9.1. Y_{stpm}^S \leq ZZ_{sg}^S, \quad \forall s, t, p, m, g | s \in S_t^T, 1 \leq m \leq n_{tp}^M$$

- Limit firing units for restricted targets:

$$9.2. \sum_s ZZ_{sg}^S \leq n_g^{FU}, \quad \forall g \in G^R$$

- Allocate all (or none) of the tasks in restricted targets:

$$9.3. U_{tpm} = U_{t'pm}, \quad \forall t, t', g \left| \begin{array}{l} n_{tp}^M > 0, \text{ for } p \in \{ "PR", "GH" \}, m = "1" \\ t, t' \in T_g^G, t \neq t', g \in G^R \end{array} \right.$$

## 10. Non-separable Tasking

- Sets

$P_t^{NS}$ , subset of task-parts in task  $t$  that are not separable;  $P_t^{NS} \subset P$

- Decision Variables

$NS_{stp}^S$ , equals 1 if all missile requirements for non-separable task-part  $p$  of task  $t$  are assigned to ship  $s$ , 0 otherwise

- Formulation

Additional constraints:

- Ensure the allocation of all missiles for a non-separable task-part to the same firing unit:

$$10.1. \quad \sum_{m=1}^{n_{tp}^M} Y_{stpm}^S = n_{tp}^M NS_{stp}^S, \quad \forall s, t, p \mid s \in S_t^T, p \in P_t^{NS}, n_{tp}^M \geq 2$$

## 11. TSC Specifications

- Sets

$TSC^S$ , subset of six-tuples  $(s, h, w, t, p, m)$  such that the TSC specifies that task-part missile  $(t, p, m)$  must be allocated to weapon type  $w$  in half-module  $h$  on ship  $s$ .

- Formulation

Additional constraints:

- TSC specifications:

$$11.1. \quad X_{shwtpm}^S = 1, \quad \forall (s, h, w, t, p, m) \in TSC^S$$

## 12. Submarines

Most sets and data for ships have an equivalent notation for submarines. Model constraints can be extended in order to incorporate submarines. Some of these extensions are immediate. Others, however, require more elaboration due to differences in specifications for ships and submarines. Here is an example:

- Indices

$B$ , set of submarines;  $b \in B$

- Sets

$B_t^T$ , subset of geo-feasible submarines for task  $t$ ;  $B_t^T \subset B$

- Data

$n_{bw}^{BT}$ , number of weapons of type  $w$  in the torpedo room of submarine  $b$

$m_b^{BT}$ , number of torpedo tubes on submarine  $b$

- Decision Variables

$X_{bwtpm}^{BT}$ , equals 1 if a weapon of type  $w$  in the torpedo room of submarine  $b$  is used to allocate missile  $m$  of task-part  $p$  in task  $t$ , 0 otherwise

- Formulation

Additional constraints:

- Avoid near-simultaneous task allocation to torpedo tubes (similar to (1.5) for ships):

$$12.1. \sum_{w|n_{bw}^{BT}>0} \sum_{t \in T_i^T \left| \begin{array}{l} b \in B_t^T, \\ w \in W_t^T \end{array} \right.} \sum_p \sum_{m=1}^{n_{tp}^M} X_{bwtpm}^{BT} \leq m_b^{BT}, \quad \forall b, i$$

- Binary variables

$$12.2. X_{bwtpm}^{BT} \in \{0, 1\}, \quad \forall b, w, t, p, m \left| b \in B, w \in W_t^T, n_{bw}^{BT} > 0, t \in T, p \in P, 1 \leq m \leq n_{tp}^M \right.$$

Accordingly, the formulation for goals and other constraints must be extended in order to incorporate submarines. The complete formulation is shown in the Appendix.

### 13. Levels of Achievement for Goals

When goal  $k > 1$  is being optimized, it is possible to specify a target level for all higher priority goals,  $k' = 1, \dots, k - 1$ .

- Data

$\varepsilon_k$ , percentage of deviation (from the best possible) allowed for goal  $k$  when a less priority goal is considered in the objective function  
 $\gamma_k$ , direction of goal  $k$ :  $\gamma_k = 1$  if maximization,  $\gamma_k = -1$  if minimization

- Formulation

Additional constraints:

- Force levels of achievement for goals:

$$13.1. Z_{k'} \leq Z_{k'}^*(1 + \varepsilon_k), \quad \forall k' | k' < k, \gamma_{k'} = -1$$

$$13.2. Z_{k'} \geq Z_{k'}^*(1 - \varepsilon_k), \quad \forall k' | k' < k, \gamma_{k'} = 1$$

### 14. Combined Objective Function

As an alternative to hierarchical optimization, we may optimize a combined (weighed) objective function of all goals.

- Data

$\omega_k$ , weight given to goal  $k$  when a combined-goal objective is considered, where  $\omega_1 \gg \omega_2 \gg \dots \gg \omega_8$ .

- Decision Variables

$Z_o$ , combined-goal objective

- Formulation

Additional constraints:

- Combined goal: Minimize a weighed average of the goals  $k \in K$ :

$$14.1. \quad Z_0^* = \min Z_0, \text{ where } Z_0 = \sum_k -\gamma_k \omega_k z_k$$

### 15. Multi-Objective Formulation

Appendix A contains one formulation (A.1)-(A.73) that is the union of all the previous formulations. We consider two formulations for our multi-objective problem:

- Hierarchical formulation (H): For each  $k \in K$ , solve the following sequence of models:

$$(H_k): \quad Z_k^* = \min(\text{or max}) Z_k \\ \text{s.t. (A.1) - (A.73)}$$

Remark: To be more precise, we write all these models as the following minimization problems:

$$(H_k): \quad Z_k^* = -\gamma_k \min -\gamma_k Z_k \\ \text{s.t. (A.1) - (A.73)}$$

- Combined-goal (sometimes referred to as weighed-sum) formulation (C):

$$(C): \quad Z_o^* = \min Z_o \\ \text{s.t. (A.1) - (A.45)} \\ \text{(A.48) - (A.73)}$$

## C. PRELIMINARY VALIDATION OF THE MODEL

In this section, we test our formulation with a small case from “NPS Report on TLAM Heuristic” [2003] in which the heuristic algorithm fails to obtain the optimal solution. As we will see, our model provides the optimal solution by using both the

Hierarchical and Combined-goal formulations. Figures 12 and 13 show the task specifications and firing units for the case, respectively.

TASKS SPECIFICATIONS							
Target	Task	Conflict Set	# of PR	# of RS	# of BU	# of GH	M <sup>3</sup> List
T1	T1-1	1	2	1	1	0	III-C
T2	T2-1	1	1	0	0	0	II-C,III-C
	T2-2	2	1	0	0	0	III-C
	T2-3	2	1	0	0	0	II-C,III-C

Figure 12. Task Specifications for Validation Case.

We have two targets and four tasks. Tasks “T1-1” and “T2-1” are near-simultaneous and the same occurs with tasks “T2-2” and “T2-3.” The M<sup>3</sup> list indicates the type of weapons that can be used for each task.

FIRING UNITS (Ships)				
FU1				
H1	III-C	III-C	II-C	II-C
H2	III-C	III-C	II-C	II-C
FU2				
H1	III-C	III-C	III-C	III-C
H2	III-C	III-C	III-C	III-C
H3	III-C	III-C	II-C	II-C

Figure 13. Firing Units for Validation Case.

There are two ships: The first one has two half-modules with III-C and II-C TLAMs, while the second one has three half-modules.

The solution provided by the heuristic algorithm (as reported in “NPS report on TLAM heuristic” [2003]) is displayed in Figure 14. It leaves task “T1-1” unmet.

NON-OPTIMAL ALLOCATION					
Target	Task	Task-Part	Firing Unit	Half-Module	Weapon
T1	T1-1	PR1	UNMET		
		PR2	UNMET		
		RS1	UNMET		
		BU1	UNMET		
T2	T2-1	PR1	FU2	H3	III-C
	T2-2	PR1	FU2	H3	III-C
	T2-3	PR1	FU2	H1	III-C

Figure 14. Non-Optimal Solution of Validation Case.

The heuristic solution cannot allocate task “T1-1.”

Two solutions to this problem are displayed in Figure 15 and Figure 16, achieved by the Hierarchical formulation and the Combined-goal formulation, respectively. As we will see, the first one is optimal, whereas the second one is near-optimal.

OPTIMAL SOLUTION					
Target	Task	Task-Part	Firing Unit	Half-Module	Weapon
T1	T1-1	PR1	FU2	H1	III-C
		PR2	FU2	H3	III-C
		RS1	FU2	H2	III-C
		BU1	FU1	H2	III-C
T2	T2-1	PR1	FU1	H1	II-C
	T2-2	PR1	FU2	H1	III-C
	T2-3	PR1	FU2	H3	II-C

Figure 15. Optimal Solution of Validation Case Obtained with the Hierarchical Formulation.

OPTIMAL SOLUTION					
Target	Task	Task-Part	Firing Unit	Half-Module	Weapon
T1	T1-1	PR1	FU2	H1	III-C
		PR2	FU2	H2	III-C
		RS1	FU2	H3	III-C
		BU1	FU1	H1	III-C
T2	T2-1	PR1	FU1	H2	II-C
	T2-2	PR1	FU2	H2	III-C
	T2-3	PR1	FU2	H3	II-C

Figure 16. Near-Optimal Solution of Validation Case Obtained with the Combined-Goal Formulation.

We have used the following weights:  $w_1 = 10^7$ ,  $w_2 = 10^6$ , ...,  $w_8 = 1$ .

Complete lists of goal achievements for both formulations are shown in Figure 17. (Remark: The Hierarchical formulation yields a better solution than the Combined-goal formulation for Goal #8 only)

Goal Priority	Goal Name	Direction	Hierarchical-goal Formulation	Combined-goal Formulation
1	Unmet Task-parts	Min	0.0	0.0
2	Employment Penalty	Min	0.0	0.0
3	Expend Firing Units	Max	0.0	0.0
4	Level Firing Units	Min	1.0	1.0
5	Primary Spread	Max	0.0	0.0
6	Back-up Spread	Max	0.0	0.0
7	M <sup>3</sup> List Position	Min	25.0	25.0
8	Residual Salvo	Max	10.0	0.0

Figure 17. Goal Achievements for Validation Case.

Both Hierarchical and Combined-goal formulations find satisfying solutions, but the solution obtained by the Hierarchical formulation is superior to that of the Combined-goal formulation for the “Salvo Capability” goal.

Further validation of our formulations is extensively analyzed in Section III.

#### D. SOLVING TLAM ALLOCATION MODEL USING FIX-AND-RELAX

In this section, we describe our use of a general-purpose methodology, Fix-and-Relax (F&R) [Dillenberger et al., 1994], to tackle some of the large-scale cases that we present in Section III. (Similar techniques were used by Brown et al. [1987] under the name of “Cascading.”) Branch-and-Bound (B&B) [Wolsey, 1998] can become inefficient to solve MIPs when the numbers of integer variables is large. In order to reduce the computational burden of solving our model with B&B, we apply F&R, which involves solving a number of sub-problems, each one with smaller complexity than the original problem. In our model, we do not implement F&R for all of the binary variables, only those from which the other binary variables are derived ( $X_{shwtpm}^s \in \{0,1\}$ ,  $X_{bwtpm}^{BT} \in \{0,1\}$  and  $X_{bwtpm}^{BC} \in \{0,1\}$ ).

For example, using the combined goal formulation, problem (C) can be rewritten as follows:

$$(C) : \min f(x, y, u)$$

$$\text{s.t.} \begin{cases} x \in \{0, 1\}^N \\ y \in \{0, 1\}^M \\ u \geq 0 \\ x, y, u \in XYU \end{cases}$$

where  $x$  is an  $N$ -dimensional vector comprised of (see Appendix A):

$$X_{shwtpm}^S, \text{ for } s \in S, 1 \leq h \leq m_S^H, w \in W_t^T, t \in T, n_{shw}^S > 0, p \in P, 1 \leq m \leq n_{tp}^M,$$

$$X_{bwtpm}^{BT}, \text{ for } b \in B, w \in W_t^T, n_{bw}^{BT} > 0, t \in T, p \in P, 1 \leq m \leq n_{tp}^M, \text{ and}$$

$$X_{bwtpm}^{BC}, \text{ for } b \in B, w \in W_t^T, n_{bw}^{BC} > 0, t \in T, p \in P, 1 \leq m \leq n_{tp}^M,$$

while  $y$  is the  $M$ -dimensional vector of the remaining binary variables, which appear in constraints (A.48)-(A.60) in the original model. In this model,  $u$  is the vector of all continuous variables, which appear in constraints (A.61)-(A.73).  $f(x, y, u)$  is our linear objective function. The constraints represented by  $x, y, u \in XYU$  are (A.1)-(A.45) and (A.48)-(A.73) for the (C) model or (A.1)-(A.73) for each of the  $(H_k)$  models.

Let  $x_1, \dots, x_n$  be the components of  $x$  where  $N$  is the total number of binary variables to which we implement F&R, that is,  $x_j$  in our abbreviated formulation is representing a variable of type  $X_{shwtpm}^S$  or  $X_{bwtpm}^{BT}$  or  $X_{bwtpm}^{BC}$ . Let  $V = \{1, \dots, N\}$  be the set of indices for these variables, and  $V_1, \dots, V_n$  be a partition of the set  $V$ . That is,  $V_i \subseteq V, \forall i = 1, \dots, n, V = \bigcup_{i=1}^n V_i$ , and  $V_i \cap V_{i'} = \emptyset, \forall i, i' = 1, \dots, n | i \neq i'$ . The cardinality of each  $V_i$  is denoted  $|V_i| = N_i$ , so  $N = \sum_{i=1}^n N_i$ .



Now, we rewrite our (C) model as:

$$(C): \min f(x, y, u)$$

$$\text{s.t.} \begin{cases} x_j \in \{0, 1\}, \forall j \in V_i, i = 1, \dots, n \\ y \in \{0, 1\}^M \\ u \geq 0 \\ x, y, u \in XYU \end{cases}$$

F&R approximates the solution to this problem by solving a sequence of  $n$  mixed-0-1 sub-problems (hereafter called *stages*), denoted as  $(C^r)$ , for  $r = 1, \dots, n$ .  $(C^r)$  can be written as follows:

$$(C^r): \min_{\substack{(x, y, u) \in XYU \\ y \in \{0, 1\}^M \\ u \geq 0}} f(x, y, u)$$

$$\text{s.t.} \begin{cases} x_j = \hat{x}_j, \forall j \in V_i, i = 1, \dots, r-1 \\ x_j \in \{0, 1\}, \forall j \in V_r \\ x_j \in [0, 1], \forall j \in V_i, i = r+1, \dots, n \end{cases}$$

where the values  $\hat{x}_j$  for  $j \in V_i, i = 1, \dots, r-1$  in stage  $r > 1$  are those obtained from the solution of problems  $(C^1), \dots, (C^{r-1})$ , respectively.  $(C^r)$  can be solved efficiently, since only a reduced subset of 0-1 integer variables must be determined at each stage  $r$ .

We select our variable partition by grouping tasks. We create  $n$  groups of tasks,  $T_1, \dots, T_n$ , such that  $\bigcup_{i=1}^n T_i = T$ .  $V_i$  is the set of variables indices associated with tasks in  $T_i$ , that is, with variables  $X_{shwtpm}^s$ ,  $X_{bwtpm}^{BT}$ ,  $X_{bwtpm}^{BC}$  for  $t \in T_i$ .

We begin our implementation by relaxing the binary constraints for the  $x$ -variables excluding the ones associated with tasks in  $T_1$ , which are forced to be binary. After the initial solution obtained from  $(C^1)$ , these binary variables are fixed in

(C<sup>2</sup>). Then, we force a new set of variables (associated with tasks in  $T_2$ ) to be integer, and repeat the process until we solve the last stage model (C<sup>n</sup>).

Model (C<sup>r</sup>) for  $r=1,...,n$  can be stated as:

$$(C^r): \min f(x, y, u)$$

$$\text{s.t.} \left\{ \begin{array}{l} \text{(A.1) to (A.45) and (A.48) to (A.73)} \\ X_{shwtpm}^S = \hat{X}_{shwtpm}^S, \text{ for } s \in S, 1 \leq h \leq m_S^H, w \in W_t^T, n_{shw}^S > 0, p \in P, 1 \leq m \leq n_{tp}^M, t \in T_i, i < r \\ X_{bwtpm}^{BT} = \hat{X}_{bwtpm}^{BT}, \text{ for } b \in B, w \in W_t^T, n_{bw}^{BT} > 0, p \in P, 1 \leq m \leq n_{tp}^M, t \in T_i, i < r \\ X_{bwtpm}^{BC} = \hat{X}_{bwtpm}^{BC}, \text{ for } b \in B, w \in W_t^T, n_{bw}^{BC} > 0, p \in P, 1 \leq m \leq n_{tp}^M, t \in T_i, i < r \\ X_{shwtpm}^S \in \{0,1\}, \text{ for } s \in S, 1 \leq h \leq m_S^H, w \in W_t^T, n_{shw}^S > 0, p \in P, 1 \leq m \leq n_{tp}^M, t \in T_r \\ X_{bwtpm}^{BT} \in \{0,1\}, \text{ for } b \in B, w \in W_t^T, n_{bw}^{BT} > 0, p \in P, 1 \leq m \leq n_{tp}^M, t \in T \\ X_{bwtpm}^{BC} \in \{0,1\}, \text{ for } b \in B, w \in W_t^T, n_{bw}^{BC} > 0, p \in P, 1 \leq m \leq n_{tp}^M, t \in T \\ X_{shwtpm}^S \in [0,1], \text{ for } s \in S, 1 \leq h \leq m_S^H, w \in W_t^T, n_{shw}^S > 0, p \in P, 1 \leq m \leq n_{tp}^M, t \in T_i, i > r \\ X_{bwtpm}^{BT} \in [0,1], \text{ for } b \in B, w \in W_t^T, n_{bw}^{BT} > 0, p \in P, 1 \leq m \leq n_{tp}^M, t \in T_i, i > r \\ X_{bwtpm}^{BC} \in [0,1], \text{ for } b \in B, w \in W_t^T, n_{bw}^{BC} > 0, p \in P, 1 \leq m \leq n_{tp}^M, t \in T_i, i > r \end{array} \right.$$

Let  $Z^*(C)$  represent the optimal objective function value for model (C). Since (C<sup>1</sup>) is a relaxation of (C), the optimal objective function value of this model,  $Z^*(C^1)$  is a lower bound on  $Z^*(C)$ . Similarly, since (C<sup>n</sup>) is a restriction of (C), its objective function value,  $Z^*(C^n)$  is an upper bound on  $Z^*(C)$ , and provides us with a feasible solution to problem (C).

### **Basic F&R Algorithm for Model (C)**

**Input:** Partition  $V_1, ..., V_n$ , where each  $V_i$  contains exactly all the  $x$ -variables associated with period tasks in set  $T_i$ :

$$V_i = \{ (s, h, w, t, p, m) \text{ for } X_{shwtpm}^S, (b, w, t, p, m) \text{ for } X_{bwtpm}^{BT} \text{ and } X_{bwtpm}^{BC}, \text{ for } t \in T_i \}, \forall i = 1, \dots, n$$

**Step 1:** Set  $i = 1$  and solve  $(C^i)$

If  $(C^i)$  is infeasible, STOP: “Problem  $(C)$  is infeasible.” Otherwise, set  $\underline{Z}^*(C) = Z^*(C^1)$ .

**Step 2:** If  $i = n$ , set  $\overline{Z}^*(C) = Z^*(C^i)$  and STOP: “Problem  $(C)$  is feasible.”

Otherwise, increase  $i$  by 1.

**Step 3:** Solve  $(C^i)$  (see remark below). If  $(C^i)$  is feasible, STOP: “Problem  $(C)$  status is unknown.”

Otherwise, go back to Step 2.

**Output:**  $(C^i)$  status “Infeasible,” “Feasible” or “Unknown.” If status is “Feasible,”  $\underline{Z}^*(C)$  and  $\overline{Z}^*(C)$  are a lower and an upper bound, respectively, on the optimal solution to  $(C)$ .

Remark: If all variables in  $V_i$  are integer after solving  $(C^{i-1})$ , then this solution is already optimal to  $(C^i)$

This basic F&R algorithm could fail to provide an answer to our problem if  $(C^1)$  is feasible but, at some subsequent stage  $i > 1$ , the related problem  $(C^i)$  becomes infeasible. The reason of that situation is that we cannot determine whether the source of infeasibility is that the  $(C)$  is actually integer-infeasible (but continuous-feasible), or that the cascade fixing procedure, which works with estimates of the true optimal values of the variables, makes  $(C^i)$  infeasible unnecessarily. We can make provisions to avoid this situation by creating bigger groups of tasks (see Escudero and Salmeron [2002]). As we will show in Chapter III, in the test cases that require us to implement F&R, this situation did not occur.

A similar F&R algorithm for our  $(H)$  formulation has been implemented. In this case, the partition  $V_i$  must be specified for each goal  $k \in K$ .

THIS PAGE INTENTIONALLY LEFT BLANK

### III. IMPLEMENTATION AND COMPUTATIONAL RESULTS

In this chapter, we describe some details of our implementation and present results for our (H) and (C) models on a set of test cases drawn from Kirk [1999] and Arnold [2000].

#### A. OVERVIEW

We implement our models in GAMS [Brooke et al. (1996)] (version 2.0.8.3 with Revision 138 module) using the CPLEX solver [GAMS-CPLEX (2004)] (version 9.0.0) on a Dell Computer Precision 340 Pentium-4, at 2.8 GHz with 1 GB of random access memory.

The source code is currently divided into 11 modules comprising over 1,000 lines of code. All the data is handled separately in multiple data files. For specific details on the use of the code, see TLAM Formulation and Code [2004] or upcoming updates of that document. Next, we explain some of the details specified through our data files, and show examples of how to use them.

##### 1. Strategy File

The “Strategy” Data File (Figure 18) allows the user to specify a maximum solution time, the number of F&R stages for the (C) formulation, and whether default weights for the (C) formulation must be overwritten by customized weights (specified separately) or not.

The default weight for the goal in the  $k$ -th position on the goal priority list is  $10^{8-k}$ . Since there are eight goals, the highest priority goal on the list will receive a weight of  $10^7$ , whereas the weight for the lowest priority goal is 1. These defaults can be overwritten by the user.

\* Parameters for the strategy for CASE

```
* # of F&R stages when a weighed objective function is used
Num_FR_stages 1

* Overwrite default weights
Overwrite_weights 0

* Maximum time (seconds)
Maxtime 600
```

Figure 18. Strategy Data File.

This file contains user's strategy information to solve the problem. There are three fields. The first two (Num\_FR\_stages and Overwrite\_weights) apply only to the case where a weighed objective function is used. The last field (Maxtime) applies to both hierarchical and combined-goal formulations.

It is important to remark that the use of adequate goals is fundamental to achieve the desired results. Determining weights is a critical task, since these weights must ensure that no allocation can ever produce a lower priority goal to outweigh a higher priority goal. Our program allows the user to manipulate the weights in order to avoid that problem. But, the validity of the solutions depends on the extent to which the operator can determine appropriate weights for each problem. This involves considering possible ranges of variation for all goals and potential sources of numerical instability. These are two examples of why the selection of weights needs to be carried out carefully:

- Suppose the first goal units are “number of unmet task-parts” and the second goal units are “penalty units for firing unit engagement in the allocation.” Furthermore, assume that there is only one ship available and one task-part to be allocated. If the employment penalty of the ship is 100, the default weights ( $w_1=10^7$ ,  $w_2=10^6$ ) will not cause the desired effect: one unmet task-part has a cost of  $10^7 \times 1 = 10^7$ , whereas engaging the ship in the allocation has a cost of  $10^6 \times 100 = 10^8$ . Consequently, we would conclude that it is preferable not to allocate the task-part, contradicting our initial objective of giving more priority to the allocation of all tasks. To avoid this problem, it is recommended to normalize the objective function coefficients, when possible, for example, we could have used an employment penalty equal to one, instead of 100. In other cases, it may be necessary to customize all of the weight values.
- Another factor to consider is the optimality tolerance used to solve the models. The default value is 0.01 (1%) (this can also be overwritten in by the user). This provokes an early termination of the solving process (Branch-and-Bound) when the incumbent solution is proven to be within the specified tolerance from optimality. Thus, objectives with lower weights may be overlooked because changes in them cannot

provoke changes in the objective function that offset the tolerance level. An example of this is our primary validation example in Section II-D (see Figure 17). There, the lowest priority goal optimal value is 10.0 (found by the (H) formulation). However, when this goal is combined with the others, the value achieved is 0.0. In practice, this may happen even if the optimality tolerance is set to 0%, because of rounding errors in the solver.

The `Maxtime` field is the allowed time to solve the model without including the generation and data management time, which may not be negligible in some cases. Unfortunately, GAMS suffers from enormous overhead for model generation, because, for example, it is not able to use previously generated models for the same problem, with perhaps minor changes (such as changing an objective function coefficient, or adding an additional constraint). Thus, each model in the sequential process to solve the hierarchical formulation needs to be regenerated, and the same occurs with each model for the F&R variable-partition procedure. While we realize that, at this moment, overhead time (as part of the total time) is of interest, we also understand that much of this time could be avoided if, for example, we used the native interfaces for the different solvers (e.g., CPLEX callable library, OSL EKK-callable library, etc.), instead of using GAMS to create the models.

Once the `Maxtime` is established, the following time allocation is made for the different models:

- Combined objective function with one F&R stage:

In this case, all of `Maxtime` seconds are allocated to solve the model.

- Hierarchical objective function with one F&R stage per goal:

We allocate  $remaining\ time \times \left( 0.5 + \frac{1}{\# \text{ of remaining goals} + 1} \right)$  seconds to the incumbent goal. For example, if `Maxtime` is 600 seconds and there are eight goals, then 367 seconds is the allotted time to solve the first goal. If it can be solved before that time limit, for example, after 200 seconds, then there are 400 seconds left for the remaining seven goals. Thus, the second goal receives 250 seconds, and so forth.

- Combined objective function with  $n > 1$  F&R stages:

We allocate  $remaining\ time \times \left(0.5 + \frac{1}{\#\ of\ remaining\ stages + 1}\right)$  seconds to the incumbent stage. The idea is the same as the one explained in the above bullet.

- Hierarchical objective function with  $n_k$  F&R stages for goal  $k$ , where  $n_k > 1$  for some  $k \in K$ .

We allocate  $t_k = remaining\ time \times \left(0.5 + \frac{1}{\#\ of\ remaining\ goals + 1}\right)$  seconds to the incumbent goal,  $k$ .

Then, we allocate  $remaining\ time_k \times \left(0.5 + \frac{1}{\#\ of\ remaining\ stages_k + 1}\right)$  seconds to the incumbent stage within the goal, where  $remaining\ time_k$  is initialized to  $t_k$  (total allotted time for the goal), and stages are for goal  $k$ .

## 2. Goal Parameters File

Another important file specifies goal data (see Figure 19):

* Goal Parameters				
	Goal_Priority	Max_Deviation	Num_FR_stages	Goal_weight
UNMET_TP	1	.20	1	1E7
EMP_PEN	2	.20	1	1E5
EXPEND_FU	3	.20	1	1E4
LEVEL_FU	4	.20	1	1E3
PR_SPREAD	5	.20	1	1E2
BU_SPREAD	6	.20	1	1E1
MCUBE_POS	7	.20	1	1E0
SALVO_CAP	8	.20	1	1E-1

Figure 19. Goal Parameters Data File.

This file provides flexibility in the way the model is set up. There are four columns of data for each goal. The first field (Goal\_Priority) applies to both the (C) and (H) formulations. The second and third fields (Max\_Deviation and Num\_FR\_stages, respectively) apply to the hierarchical formulation. The last field (Goal\_weight) applies to the combined-goal formulation.

The Goal\_Priority field must contain a list of all integers from one through eight, where each number must be entered exactly once. The number indicates the priority of the goal and will be used for the following purposes:



- If the combined-goal is being used, a default weight equal to  $10^{8-k}$  will be allocated for the goal whose priority is precisely  $k$ . These weights will be used in the combined objective function unless otherwise specified by the user by means of the `Overwrite_weights` flag (specified in the aforementioned Strategy Data File).
- If a hierarchical formulation is being used, the list causes the highest priority goal (i.e., the goal whose priority is equal to one) to be optimized first, then the second (subject to a restriction on the first), and so forth.

The `Max_Deviation` field for a goal whose priority is  $k < 8$  specifies the level of flexibility in the achievement required for this goal in the (H) formulation. It sets the maximum deviation from the goal's optimum when any lower priority goal  $k' > k$  is being optimized. For example, using the data in Figure 19, if the optimal solution for goal #1 is 100, then goal #2 will be optimized subject to a maximum of 120 for goal #1. Remark: The deviation must be specified in absolute value because the direction for each goal (min. or max.) is already implemented in the code. Thus, if goal #1 were a maximization goal, then goal #2 would have been optimized subject to a minimum of 80 for goal #1.

The `Num_FR_stages` field contains the number of F&R stages for each goal. If this number is set to one, the optimization problem associated with that goal will be solved in one single pass. It is anticipated that some goals (such as “Spread Primary Task-Parts”) may require more computational work than others, and it is therefore advised to specify more than one F&R stage for these goals in large cases.

The `Goal_Weight` field is only used when the user solves the combined-goal formulation with customized weights. The `Overwrite_weights` flag (in the `Strategy.dat` file) must be set to “1.” Then, the `Goal_Weight` field can be used to specify new weights for each goal, which will overwrite the default ones.

### **3. Output Files**

The information of goal achievement and time used depends on the formulation used.

Total solve time:	0.3 sec.
G 1: UNMET_TP (min)	0.0
G 2: EMP_PEN (min)	1.0
G 3: EXPEND_FU (max)	0.0
G 4: LEVEL_FU (min)	16.0
G 5: PR_SPREAD (max)	1.0
G 6: BU_SPREAD (max)	1.0
G 7: MCUBE_POS (min)	25.0
G 8: SALVO_CAP (max)	36.0
OBJ_VAL_W :	1159114.00
OBJ_EST_W :	1158614.00

Figure 20. Goal Achievement Results: Combined-Goal Formulation

In the combined-goal results, the output (see Figure 20) shows:

- The total solve time
- The objective achieved for each goal
- The combined-goal achievement and a lower bound on the best possible result.  
Remark: the combined-goal formulation is stated as a minimization model, see Section II.B.

Total solve time:		1.9 sec.									
*****											
MULTI-OBJECTIVE TABLE											
*****											
	G 1	G 2	G 3	G 4	G 5	G 6	G 7	G 8	Time	(OBJ_VAL_H)	(OBJ_EST_H)
	(min)	(min)	(max)	(min)	(max)	(max)	(min)	(max)	(sec)		
G 1: UNMET_TP	0.0	1.0	0.0	20.0	0.0	0.0	29.0	0.0	0.2	0.00	0.00
G 2: EMP_PEN	0.0	1.0	0.0	20.0	0.0	0.0	29.0	0.0	0.2	1.00	1.00
G 3: EXPEND_FU	0.0	1.0	0.0	20.0	0.0	0.0	29.0	0.0	0.3	0.00	0.00
G 4: LEVEL_FU	0.0	1.0	0.0	16.0	0.0	0.0	29.0	0.0	0.3	16.00	16.00
G 5: PR_SPREAD	0.0	1.0	0.0	18.0	2.0	0.0	29.0	0.0	0.2	2.00	2.00
G 6: BU_SPREAD	0.0	1.0	0.0	18.0	2.0	0.0	29.0	0.0	0.3	0.00	0.00
G 7: MCUBE_POS	0.0	1.0	0.0	18.0	2.0	0.0	25.0	0.0	0.2	25.00	25.00
G 8: SALVO_CAP	0.0	1.0	0.0	18.0	2.0	0.0	25.0	36.0	0.2	36.00	36.00

Figure 21. Goal Achievement Results: Hierarchical Formulation.

When the (H) formulation is used, the output (see Figure 21) displays:

- The total solve time
- The objective achieved for each goal at each level of the hierarchical-optimization process
- The time needed to solve each goal
- A bound for the best possible value for each goal

For example, when we solve for the first goal (minimization), we obtain zero for that goal and other non-optimal values for the other goals (see the line headed “G 1”). This is clearer, for example, by looking at goals #3 and number #4. When goal #3 is

optimized, the result value achieved for goal #3 is zero whereas it is 20 for goal #4. Then, when we optimize goal #4, we are able to reduce this value to 16. Moreover, we know this value is optimal because the bound (last column value) indicates that the best estimate is precisely 16. However, when we optimize goal #5, which improves its value from zero to two, we notice that goal #4 worsens from 16 to 18. This is due to the fact that we have allowed a maximum of 20% deviation (see Figure 19) from its optimal for this goal.

Of course, the last row refers to the last goal that is optimized and indicates the values achieved for all of the goals after the optimization process is completed. As we can see, these values slightly differ from those achieved with the weighed objective function, due to the degree of flexibility allowed in some goals. If an optimal solution is required, then the `Max_Deviation` parameter must be set to 0.00 for all goals, and the optimality tolerance must be set to 0.00 as well.

Figure 24 is an example of allocation output. The example shows, for example, two PR task-parts in task “T1-1” in target “T1” allocated to two III-C weapons in half-modules #1 and #2 on ship DDG72. Also, we can see that submarine SSN775 is engaged in the allocation; in particular, two II-C weapons and one III-C weapon from the torpedo room are used.

***** TASKING ALLOCATION *****								
G	T	P	M	S	H	B	T/C	W
T1	T1-1	PR	TPM1	DDG72	h2			III-C
			TPM2	DDG72	h1			III-C
		RS	TPM1	DDG72	h3			III-C
		BU	TPM1	CG73	h2			III-C
T2	T2-1	PR	TPM1			SSN755	T	II-C
	T2-2	PR	TPM1			SSN755	T	III-C
	T2-3	PR	TPM1			SSN755	T	II-C

Figure 22. Allocation Results

#### 4. Example

The flexibility that our code allows is shown in the following example. We consider “Case 1” drawn from Kirk’s Thesis [1999]. Initially, we consider minimizing unmet task-parts as the highest priority goal, and maximizing residual salvo as the least priority goal. However, the TSC may require a change in the priorities, and he or she may deem maximizing residual salvo as the second priority goal (e.g., Kuykendall [1999]), to the detriment of other intermediate goals. We conduct this test by simply changing the priority list in the goal data file, as explained above. As we show in Figure 23, the achievements in some of the goals change in these two different cases.

Goal Name	Direction	Initial Goal Priority	Value	New Goal Priority	Value
Unmet Task-parts	Min	1	0.0	1	0.0
Employment Penalty	Min	2	0.0	<b>8</b>	<b>1.0</b>
Expend Firing Units	Max	3	N/A	3	N/A
Level Firing Units	Min	4	34.0	4	34.0
Primary Spread	Max	5	3.0	5	4.0
Back-up Spread	Max	6	N/A	6	N/A
M <sup>3</sup> List Position	Min	7	92.0	7	92.0
Residual Salvo	Max	8	159.0	<b>2</b>	<b>160.0</b>

Figure 23. Case 1: Changing Goal Priorities.

An important advantage of the (H) implementation is that even if we cannot prove that an incumbent solution is optimal, a bound on the best possible solution is provided. This may allow the TSC to let more solving time to optimize particular goal in a subsequent run. An example of this is “Case 5A” from Kirk [1999]. As we can see from Figure 24, we achieve 38.5 for Goal #4 before the solving time limit (set to 400 seconds for this test) is exceeded. At this point, we cannot guarantee that 38.5 is the optimal value for leveling missiles across firing units, but we know that in any case we cannot obtain a better score than 37.4.

Goal Priority	Goal Name	Direction	(H) Formulation	Best Bound
1	Unmet Task-parts	Min	0.0	0.0
2	Employment Penalty	Min	0.0	0.0
3	Expend Firing Units	Max	72.0	73.5
4	Level Firing Units	Min	38.5	37.4
5	Primary Spread	Max	4.0	4.0
6	Back-up Spread	Max	0.0	0.0
7	M <sup>3</sup> List Position	Min	594.0	586.7
8	Residual Salvo	Max	222.0	242.8

Figure 24. Case 5A: Comparison of Solution Values and Best Bounds. We limit the solving time to 400 seconds for this run. Shaded cells shows goals for which the solution achieved has not been proven optimal.

In the case of the (C) implementation, we do not have a measure of optimality for each individual goal, but for the whole combined goal that makes up the objective function.

## B. SCENARIO DESCRIPTION AND COMPUTATIONAL RESULTS

### 1. Scenario Description

Our test focuses on existing scenarios developed by previous NPS students and the NSWCCD. We also incorporate two new scenarios to our test cases. In total, we tested 23 cases, including sub-scenarios.

The first six scenarios were developed by NSWCCD and tested initially by Kirk, and then by Hodge and Arnold. Some of these test scenarios include several excursions, making a total of 11 test cases. The number of tasks in these cases ranges from 13 tasks in “Case 1” to 104 tasks in “Case 6.” All of Kirk’s 11 cases only include ships as firing units, and do not have multiple missile requirements for the same task-part, among other limitations. For additional details, consult Kirk [1999].

Arnold adds six new scenarios to test new features such as submarines, ghost tasks, targets containing multiple tasks, etc. For details, see Arnold [2000].

In addition to these twelve scenarios (and their variants), we create two more scenarios: the first, “Case 13,” is strongly based on Wingart’s most complicated case (Strike Three of RIMPAC 00-3 Exercise), which has been recreated manually by using

the information available from his thesis (as opposed to Kirk's and Arnold's scenarios for which electronic data were available). The new second scenario, "Case 14" is a hypothetical case intended to implement multiple specifications (described in Chapter II) in a single case, for example, submarine back-up tasking, non-separable tasking, TSC specifications, etc. "Case 13" includes 100 tasks and "Case 14" has 26 tasks.

## 2. Test Case Set Up

We test all scenarios using both the (C) and (H) formulations. Unless otherwise specified, we use the following set of parameters to run all test cases:

- (H) formulation (multiple submodels, one for each goal, optimized hierarchically):
  - Optimality tolerance: 0.01 (1%) for each submodel.
  - "Maxtime" parameter: (maximum solve time for the overall process including all submodels): 3,600 seconds
  - "Num\_FR" parameter (number of F&R stages per submodel): One (i.e., solve the submodel as a whole without variable partition) for all goals.
  - "Max\_Deviation" parameter (percentage of deviation from optimal goal value): 0.01 (1%) for all the goals.
- (C) formulation (one combined objective function with weighed goals):
  - Optimality tolerance: 0.01 (1%).
  - "Maxtime" parameter: 3,600 seconds.
  - "Num\_FR" parameter (number of F&R stages): One (i.e., solve the model as a whole without variable partition).
  - Weights used:  $\omega_k = 10^{8-k}$ ,  $\forall k \in K = \{1, \dots, 8\}$

In some special cases, we will use F&R with more than one stage in order to increase the model or submodel solvability. For example, we may be applying F&R to the submodel associated with a special goal in the (H) formulation or to the model in the (C) formulation.

In other cases, we may be reducing the maximum time allotted for the overall process in order to check how this affects the quality of the incumbent solution as well as the best bound on the optimal solution if this is not achieved.

### 3. Results

We compare our results to those obtained by Kirk [1999] and Arnold [2000] (the former for applicable cases only). We omit Hodge [1999] from the comparison because his results are always outperformed by Kirk and Arnold.

Table 1 shows the results for all scenarios. The shaded cells show the key differences between our solutions and the existing ones. As explained earlier in this thesis, we display solution time as that required to solve the model, omitting overhead time caused by model generation and other data management in GAMS.

For the first six scenarios and their variants, we notice the reduction in computational time to solve our (H) and (C) formulations when compared to that reported by Kirk [1999], who produced the last available MIP formulation prior to this work. One of the most dramatic cases is “Case 5,” in which solution time decreases from 10,976 seconds for Kirk’s Hierarchical solution to 40.5 seconds for our (H) formulation.

Regarding the quality of the solutions provided by our formulations, it is the same as or better than that of Kirk and Hodge (where available), and Arnold (in all cases, except “Case 2” where we believe there is an error in the result produced by Arnold, which is also confirmed by Kirk’s result). (Remark: In some cases, such as “Case 1B” using the (H) formulation, we notice that the optimal value reported for Goal #8 is 159, whereas the optimal is 160. This and other minor discrepancies can be attributed to the optimality tolerance established at 1% for all models. In these cases, we still consider both solutions of the same quality.)

“Case 13” is similar to that of LT Wingear’s most complicated case: RIMPAC 00-3 Strike Three case, with 100 tasks [Wingear, 2001]. Because the full data set for this case was not available at the time of developing this work, we manually recreated the case using the information shown in Wingear’s thesis. Consequently, our results can be considered comparable to those of Wingear in terms of scope and case difficulty, but not necessarily on a goal by goal basis.

Finally, we tested a completely new scenario, “Case 14,” to further validate the incorporation of all problem specifications into our formulation.

Test Case	Implementation Source	GOALS								Solution Time (Seconds)
		# 1 (Min): Unassigned Task-parts	# 2 (Min): Penalty Units*	# 3 (Max): Expend Units	# 4 (Min): Deviation Level	# 5 (Max): Primary Missiles	# 6 (Max): Back-up Missiles	# 7 (Min): M <sup>3</sup> List Position	# 8 (Max): Residual Salvo	
1	Kirk (Hier. R.)	0	0	N/A	34	3	N/A	92	160	26
	Arnold (Heur.)	0	0	N/A	34	3	N/A	92	160	2
	(C) Model	0	0	N/A	34	3	N/A	92	160	0.3
	(H) Model	0	0	N/A	34	3	N/A	92	159	1.8
1A	Kirk (Hier. R.)	0	N/A	10	8.7	N/A	N/A	92	160	34
	Arnold (Heur.)	0	N/A	10	8.7	N/A	N/A	92	160	2
	(C) Model	0	N/A	10	8.7	N/A	N/A	92	160	4.3
	(H) Model	0	N/A	10	8.7	N/A	N/A	92	160	1.8
1B	Kirk (Hier. R.)	0	N/A	10	8.7	N/A	<u>2</u>	92	160	36
	Arnold (Heur.)	0	N/A	10	8.7	N/A	<u>3</u>	92	160	2
	(C) Model	0	N/A	10	8.7	N/A	<u>4</u>	92	160	0.3
	(H) Model	0	N/A	10	8.7	N/A	<u>4</u>	92	159	2.1
2	Kirk (Hier. R.)	0	0	N/A	23	3	N/A	184	157	335
	Arnold (Heur.)	0	0	N/A	13.5**	2	N/A	221	162	3
	(C) Model	0	0	N/A	23	3	N/A	184	157	0.8
	(H) Model	0	0	N/A	23	3	N/A	184	157	5.6
2A	Kirk (Hier. R.)	0	1	N/A	42.5	2	N/A	196	146	103
	Arnold (Heur.)	0	1	N/A	44.5	4	N/A	196	141	4
	(C) Model	0	1	N/A	42.5	4	N/A	186	146	0.4
	(H) Model	0	1	N/A	42.5	4	N/A	186	145	1.9
3	Kirk (Hier. R.)	0	0	N/A	15.5	3	N/A	368	<u>129</u>	1625
	Arnold (Heur.)	0	<u>1</u>	N/A	1.5	3	N/A	368	130	5
	(C) Model	0	<u>0</u>	N/A	15.5	3	N/A	368	<u>140</u>	1.5
	(H) Model	0	<u>0</u>	N/A	15.5	3	N/A	368	<u>140</u>	14.5
4	Kirk (Hier. R.)	0	1	N/A	0	4	N/A	496	127	6133
	Arnold (Heur.)	0	1	N/A	<u>16</u>	4	N/A	508	107	10
	(C) Model	0	1	N/A	<u>0</u>	4	N/A	484	127	31.7
	(H) Model	0	1	N/A	<u>0</u>	4	N/A	488	126	23.1
5	Kirk (Hier. R.)	0	0	N/A	11.4	4	N/A	558	268	10976
	Arnold (Heur.)	0	0	N/A	<u>25.4</u>	6	N/A	599	252	10
	(C) Model	0	0	N/A	<u>11.4</u>	4	N/A	558	268	6.1
	(H) Model	0	0	N/A	<u>11.4</u>	4	N/A	562	268	40.4
5A	Kirk (Hier. R.)	0	0	<u>65</u>	34	5	N/A	594	242	2356
	Arnold (Heur.)	0	0	73	40	4	N/A	596	227	10
	(C) Model	0	0	<u>73</u>	40	4	N/A	597	230	4.5
	(H) Model	0	0	<u>72</u>	38.5	4	N/A	597	241	3367.9
5B	Kirk (Hier. R.)	0	1	N/A	21.4	<u>4</u>	N/A	564	268	34340
	Arnold (Heur.)	0	1	N/A	<u>35.1</u>	5	N/A	629	240	16
	(C) Model	0	1	N/A	<u>21.4</u>	<u>5</u>	N/A	583	268	1349



Test Case	Implementation Source	GOALS								Solution Time (Seconds)
		# 1 (Min): Unassigned Task-parts	# 2 (Min): Penalty Units*	# 3 (Max): Expend Units	# 4 (Min): Deviation Level	# 5 (Max): Primary Missiles	# 6 (Max): Back-up Missiles	# 7 (Min): M <sup>3</sup> List Position	# 8 (Max): Residual Salvo	
	(H) Model	0	1	N/A	20.6	5	N/A	592	270	2328
6	Kirk (Hier. R.)	0	0	69	24	6	N/A	904	209	261
	Arnold (Heur.)	0	0	69	24	6	N/A	902	201	7
	(C) Model	0	0	78	13.5	5	N/A	892	240	22.2
	(H) Model	0	0	78	13.5	5	N/A	896	238	226.8
7	Arnold (Heur.)	0	0	N/A	34	2	N/A	92	160	2
	(C) Model	0	0	N/A	34	3	N/A	92	152	0.3
	(H) Model	0	0	N/A	34	3	N/A	92	152	1.4
7A	Arnold (Heur.)	0	0	10	34.7	N/A	N/A	104	152	5
	(C) Model	0	0	10	34.7	N/A	N/A	107	147	1.7
	(H) Model	0	0	10	34.7	N/A	N/A	107	147	1.4
7B	Arnold (Heur.)	0	0	10	8.7	N/A	3	92	160	2
	(C) Model	0	0	10	8.7	N/A	4	92	150	6
	(H) Model	0	0	10	8.7	N/A	4	92	150	1.9
8	Arnold (Heur.)	0	0	N/A	36	N/A	N/A	80	162	2
	(C) Model	0	0	N/A	36	N/A	N/A	80	162	0.3
	(H) Model	0	0	N/A	36	N/A	N/A	80	161	2
8A	Arnold (Heur.)	0	0	N/A	36	N/A	N/A	81	162	2
	(C) Model	0	0	N/A	36	N/A	N/A	80	153	0.2
	(H) Model	0	0	N/A	36	N/A	N/A	80	153	1.4
9	Arnold (Heur.)	0	0	N/A	34	3	N/A	92	160	3
	(C) Model	0	0	N/A	34	3	N/A	92	160	0.3
	(H) Model	0	0	N/A	34	3	N/A	92	160	1.8
9A	Arnold (Heur.)	0	0	N/A	34	N/A	3	92	160	3
	(C) Model	0	0	N/A	34	N/A	3	92	153	0.3
	(H) Model	0	0	N/A	34	N/A	3	92	153	1.6
10	Arnold (Heur.)	0	0	N/A	39	3	N/A	157	147	3
	(C) Model	0	0	N/A	31	3	N/A	149	152	0.6
	(H) Model	0	0	N/A	31	3	N/A	149	152	2.9
10A	Arnold (Heur.)	0	0	N/A	37	3	N/A	167	147	3
	(C) Model	0	0	N/A	31	3	N/A	149	152	0.6
	(H) Model	0	0	N/A	31	3	N/A	149	152	2.6
11	Arnold (Heur.)	7***	0	N/A	0	1	N/A	58	162	1
	(C) Model	0	0	N/A	34	3	N/A	92	160	0.3
	(H) Model	0	0	N/A	34	3	N/A	92	159	1.7
12	Arnold (Heur.)	0	0	N/A	43	7	N/A	654	256	12
	(C) Model	0	0	N/A	15.8	6	N/A	635	272	12
	(H) Model	0	0	N/A	15.8	6	N/A	641	272	46.4
13	Wing.(Heur(B))	0	N/A	N/A	49	8	6	128	168	N/A
	Wing.(Heur(W))	4	N/A	N/A	73.5	8	3	111	177	N/A

Test Case	Implementation Source	GOALS								Solution Time (Seconds)
		# 1 (Min): Unassigned Task-parts	# 2 (Min): Penalty Units*	# 3 (Max): Expend Units	# 4 (Min): Deviation Level	# 5 (Max): Primary Missiles	# 6 (Max): Back-up Missiles	# 7 (Min): M <sup>3</sup> List Position	# 8 (Max): Residual Salvo	
	Wing.(Split(B))	0	N/A	N/A	49	8	5	128	169	N/A
	Wing.(Split(W))	6	N/A	N/A	73.5	8	3	112	176	N/A
	(C) Model	0	N/A	N/A	45	6	7	448	371	7.3
	(H) Model	0	N/A	N/A	45	6	7	451	371	199
14	(C) Model	0	0	24	7.3	2	2	227	178	0.5
	(H) Model	0	0	24	7.3	2	2	227	177	2.9

Table 1. Result List.

Maximum time allowed is 3,600 seconds and optimality tolerance is 0.01 for all runs.

\*: To be consistent with reports by previous students, actual employment penalties are divided by 100.

\*\*: Arnold solution for this case appears to be incorrect.

\*\*\*: Available data for this case appears to be inconsistent with Arnold's report.

#### 4. Refinements to Tackle the Hardest Test Cases

While MIP solution quality is expected to exceed heuristics, there are a few instances in which solving the MIP may require a prohibitive computational time. It is not possible to know which cases will require an unacceptable time to be optimally solved, but we have made provisions to speed up the MIP solver.

First, we establish priority branching as follows: Half-module-related variables are branched first. Then, torpedo-tube-related variables are branched, and lastly, canister-capsule-variables. Within each of these categories, the more weapons of a given type the asset has, the higher the branching priority. For example, given  $n_{shw}^s$  as the number of weapons of type  $w$  in half-module  $h$  on ship  $s$ , the priority value given to  $X_{shwtpm}^s$  is  $1/(1+n_{shw}^s)$ . Remark: The lower the priority value, the sooner the associated variable is branched. By using these priorities, we notice that a slight improvement in computational time for some cases. (typically under 10%).

In this section we report the results obtained by (a) reducing the total solving time and (b) using F&R with several stages in order to partition our models into easier subproblems.

**a. Reducing the Solving Time**

We observe that “Case 5B” is one of the most difficult cases to solve. In fact, when we let the maximum solving time increase, this case uses up all the time available. When we decrease the maximum solving time to 600 seconds, the (H) formulation gives the same near-optimal solution as before (see Figure 25). We notice that in both cases, the best available bound for goal #4 is 11.4, while the best solution found is 20.6. It is clear that this particular problem is consuming all the available time allocated for that goal, which is dynamically computed as explained in Section A.

Goal			Maximum Time (36000 seconds)		Maximum Time (600 seconds)	
Priority	Name	Direction	(H) Formulation	Best Bound	(H) Formulation	Best Bound
1	Unmet Task-parts	Min	0.0	0.0	0.0	0.0
2	Employment Penalty	Min	1.0	1.0	1.0	1.0
3	Expend Firing Units	Max	N/A	N/A	N/A	N/A
4	Level Firing Units	Min	20.6	11.4	20.6	11.4
5	Primary Spread	Max	5.0	5.0	5.0	5.0
6	Back-up Spread	Max	N/A	N/A	N/A	N/A
7	M <sup>3</sup> List Position	Min	592.0	582.4	592.0	582.4
8	Residual Salvo	Max	270.0	271.0	270.0	271.0
Solution Time (Seconds)			21767.0		584.4	

Figure 25. Case 5B: Reducing Maximum Time.

We set the maximum solving time to 36,000 seconds and solve the case with the (H) formulation. The solution and bounds achieved are the same as those achieved when the maximum time is set to 600 seconds. Some goals such as goal #4 are using all the time allocated to solve them.

**b. Using Fix and Relax**

In Figure 26, we show the results obtained by applying F&R to our (H) formulation for “Case 5A.” Without F&R, the (H) formulation consumes almost all the available time (3,600 seconds). As in “Case 5B” above, some of the goals are using up all the allotted time. When we use F&R for selected goals (identified as complicated

goals for this case), we observe that the total solving time is reduced drastically, without compromising the quality of the solution value and the associated bounds in this case.

Goal			(H) Formulation		(H) Formulation with F&R		
Priority	Name	Direction	Value	Best Bound	# of Stages	Value	Best Bound
1	Unmet Task-parts	Min	0.0	0.0	1	0.0	0.0
2	Employment Penalty	Min	0.0	0.0	1	0.0	0.0
3	Expend Firing Units	Max	72.0	73.5	4	73.0	73.0
4	Level Firing Units	Min	38.5	37.4	4	40.0	40.0
5	Primary Spread	Max	4.0	4.0	4	4.0	4.0
6	Back-up Spread	Max	N/A	N/A	4	N/A	N/A
7	M <sup>3</sup> List Position	Min	597.0	586.7	4	597.0	592.6
8	Residual Salvo	Max	241.0	242.4	4	239.0	239.8
Solution Time (Seconds)			3367.9		316.7		

Case 5A: Results with Fix-and-Relax.

Solution quality is not compromised in this case.

## IV. CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER RESEARCH

This thesis consolidates the latest specifications for TLAM pre-designation into a revised mathematical formulation, and implements it as a multi-objective mixed-integer optimization problem. We show that our models can be optimally solved in reasonable time for most of the test cases, and in those cases where computational time is an issue, we establish mechanisms to alleviate the model complexity and achieve solutions of acceptable quality.

On the other hand, through our models we validate heuristic results for a number of test cases developed by former NPS students. Although the heuristic may fail to provide an optimal answer even for the highest priority goal, its overall performance can be considered satisfactory.

There are several aspects in which this thesis contributes to and enhances earlier work on automated TLAM predesignation. With respect to Kirk's work [Kirk, 1999], the contributions lie on:

- Reformulating multiple parts of the model, making it possible to attain the solution of problems that could not be solved before by the combined-goal formulation (C), and drastically reducing the time for the hierarchical formulation (H).
- Incorporating all new specifications, such as submarines, targets, multiple task-part requirements, etc.
- Implementing a Fix-and-Relax (F&R) variable-partition procedure to help solve the most complicated cases.
- Implementing our programs in a much more amenable fashion, where the front-end for the user is composed of a number of data files completely independent of the source code. For example, changing goal priorities is as simple as updating an array of eight numbers. All of our test cases use the same code, and all reasonable data and option customization is done at the level of data files.

With respect to Arnold's work, our (C) and (H) formulations:

- Enhance the quality of the solution obtained.
- Provide a valid bound on the optimal solution in case this is not achieved.
- Have the flexibility to arrange goals in any desired order, to surrogate certain goals to desired levels of achievement in other goals ((H) formulation), or to allocate different

weights in order to generate a variety of efficient points for the multi-objective problem ((C) formulation). These are important limitations in the heuristic.

The work in this thesis is part of an ongoing effort by students and faculty at NPS and NSWCCD. In that context, we believe that further validation of our models with real data (not available at the time of completing this thesis) is necessary.

We also realize that GAMS may not be the most appropriate environment for model generation. In some cases, the model generation and execution time is even greater than the time to solve the (already generated) model in CPLEX. This burden intensifies when multiple models (with minor differences between them) need to be solved, such as in the (H) formulation and/or when F&R is used.

## APPENDIX A. FULL-TLAM ALLOCATION MODEL

### A. NOTATION

#### 1. Indices

$S$ ,	set of ships; $s \in S$
$H$ ,	set of half-modules; $h \in H = \{1, 2, 3, \dots\}$
$B$ ,	set of submarines; $b \in B$
$G$ ,	set of targets; $g \in G$
$T$ ,	set of tasks; $t \in T$
$P$ ,	set of task-parts; $p \in P = \{PR, RS, BU, GH\}$
$M$ ,	set of task-part missile requirements; $m, m' \in M = \{1, 2, 3, \dots\}$
$W$ ,	set of weapon types; $w \in W$
$I$ ,	set of subsets near-simultaneous tasks; $i \in I$
$A$ ,	set of launch areas; $a \in A$
$K$ ,	set of goals; $k \in K = \{1, 2, \dots, 8\}$

#### 2. Sets

$T_g^G$ ,	subset of tasks for target $g$ ; $T_g^G \subset T$
$T_i^I$ ,	subset of near-simultaneous tasks in group $i$ ; $T_i^I \subset T$
$G_a^A$ ,	subset of targets which are geo-feasible from area $a$ ; $G_a^A \subset G$
$G^R$ ,	subset of targets which are designated as restricted; $G^R \subset G$
$A^{PR}$ ,	subset of areas whose primary task-parts are desired to be assigned to as many firing units as possible; $A^{PR} \subset A$
$A^{BU}$ ,	subset of areas whose back-up task-parts are desired to be assigned to as many firing units as possible; $A^{BU} \subset A$
$S^{Exp}$ ,	subset of expend ships; $S^{Exp} \subset S$
$B^{Exp}$ ,	subset of expend submarines; $B^{Exp} \subset B$
$B^{BU}$ ,	subset of submarines that are desired to be assigned to back-up task-part allocation of any task $t$ ; $B^{BU} \subset B$
$P_t^{NS}$ ,	subset of task-parts in task $t$ that are not separable; $P_t^{NS} \subset P$
$W_t^T$ ,	subset of weapons that can be used for task $t$ ; $W_t^T \subset W$

$TSC^S$ ,	subset of six-tuples $(s, h, w, t, p, m)$ such that the TSC specifies that task- part missile $(t, p, m)$ must be allocated to weapon type $w$ in half-module $h$ on ship $s$ .
$TSC^{BT}$ ,	subset of five-tuples $(b, w, t, p, m)$ such that the TSC specifies that task- part missile $(t, p, m)$ must be allocated to weapon type $w$ in the torpedo room of submarine $b$ .
$TSC^{BC}$ ,	subset of five-tuples $(b, w, t, p, m)$ such that the TSC specifies that task-part missile $(t, p, m)$ must be allocated to weapon type $w$ in the canister capsules of submarine $b$ .

### 3. Data

$n_{shw}^S$ ,	number of weapons of type $w$ in half-module $h$ on ship $s$
$n_{bw}^{BT}$ ,	number of weapons of type $w$ in the torpedo room of submarine $b$
$n_{bw}^{BC}$ ,	number of weapons of type $w$ in the canister capsules of submarine $b$
$m_s^H$ ,	number of half-modules on ship $s$
$m_b^{BT}$ ,	number of torpedo tubes on submarine $b$
$n_{tp}^M$ ,	number of missiles required for task-part $p$ of task $t$
$p_s^S$ ,	employment penalty for ship $s$
$p_b^B$ ,	employment penalty for submarine $b$
$n^{NExp}$ ,	number of non-expend firing units: $n^{NExp} =  S \setminus S^{Exp}  +  B \setminus B^{Exp} $
$a_s^S$ ,	area for ship $s$
$a_b^B$ ,	area for submarine $b$
$m_p^3$ ,	weight of task-part $p$ to match the $M^3$ list position of for task-part $p$
$m_{tw}^3$ ,	$M^3$ position of weapon $w \in W_t^T$ for task $t$
$v_w$ ,	relative value of weapon $w$ for residual salvo computation
$n_g^{FU}$ ,	maximum number of firing units to which a restricted target $g$ can be allocated
$\varepsilon_k$ ,	percentage of deviation (from the best possible) allowed for goal $k$ when a less priority goal is considered in the objective function



- $\gamma_k$ , direction of goal  $k$ :  $\gamma_k = 1$  if maximization,  $\gamma_k = -1$  if minimization
- $\omega_k$ , weight given to goal  $k$  when a combined goal objective is considered, where  $\omega_1 \gg \omega_2 \gg \dots \gg \omega_8$ .

#### 4. Derived Data and Sets

- $g_t$ , target for task  $t$  (i.e.,  $t \in T_g^G$ )
- $S_t^T$ , subset of geo-feasible ships for task  $t$ ;  $S_t^T = \{s \mid g_t \in G_{a_s}^A\}$
- $B_t^T$ , subset of geo-feasible submarines for task  $t$ ;  $B_t^T = \{b \mid t \in G_a^A \text{ For } a = a_b^B\}$

#### 5. Decision Variables

- $X_{shwtpm}^S$ , equals 1 if a weapon of type  $w$  in half-module  $h$  of ship  $s$  is used to allocate missile  $m$  of task-part  $p$  in task  $t$ , 0 otherwise
- $X_{bwtpm}^{BT}$ , equals 1 if a weapon of type  $w$  in the torpedo room of submarine  $b$  is used to allocate missile  $m$  of task-part  $p$  in task  $t$ , 0 otherwise
- $X_{bwtpm}^{BC}$ , equals 1 if a weapon of type  $w$  in the canister capsules of submarine  $b$  is used to allocate missile  $m$  of task-part  $p$  in task  $t$ , 0 otherwise
- $Y_{stpm}^S$ , equals 1 if missile  $m$  of task-part  $p$  in task  $t$  is allocated to ship  $s$ , 0 otherwise
- $Y_{btpm}^B$ , equals 1 if missile  $m$  of task-part  $p$  in task  $t$  is allocated to submarine  $b$ , 0 otherwise
- $U_{tpm}$ , equals 1 if task  $t$  is unmet, 0 otherwise
- $ZZ_{sg}^S$ , equals 1 if ship  $s$  is engaged in any tasking in target  $g$ , 0 otherwise
- $Z_s^S$ , equals 1 if ship  $s$  is used, 0 otherwise
- $ZZ_{bg}^B$ , equals 1 if submarine  $b$  is engaged in any tasking in target  $g$ , 0 otherwise
- $Z_b^B$ , equals 1 if submarine  $b$  is used, 0 otherwise
- $E^S$ , total number of weapons allocated to primary task-parts from expend ships
- $E^B$ , total number of weapons allocated to primary task-parts from expend submarines

$NS_{stp}^S$ ,	equals 1 if all missile requirements for non-separable task-part $p$ of task $t$ are assigned to ship $s$ , 0 otherwise
$NS_{btp}^B$ ,	equals 1 if all missile requirements for non-separable task-part $p$ of task $t$ are assigned to submarine $b$ , 0 otherwise
$RN_s^S$ ,	number of missiles on non-expend ship $s$ that are not allocated to a primary task-part
$RN_b^B$ ,	number of missiles on non-expend submarine $b$ that are allocated to a primary task-part
$RAVG$ ,	average of the residual number of missiles across non-expend firing units after allocation
$RD_s^S$ ,	difference between the residual number of missiles on ship $s$ and $RAVG$
$RD_b^B$ ,	difference between the residual number of missiles on submarine $b$ and $RAVG$
$PR_s^S$ ,	equals 1 if ship $s$ is assigned to allocated any primary task-part, 0 otherwise
$PR_b^B$ ,	equals 1 if submarine $b$ is assigned to allocated any primary task-part, 0 otherwise
$BU_s^S$ ,	equals 1 if ship $s$ is assigned to any back-up task-part, 0 otherwise
$BU_b^B$ ,	equals 1 if submarine $b$ is assigned to any back-up task-part, 0 otherwise
$D_{shw}^S$ ,	equals 1 if one or more weapons of type $w$ remain on ship $s$ in half-module $h$ after firing all primary task-part missiles, 0 otherwise
$D_{bw}^{BT}$ ,	number of weapons of type $w$ remain in the torpedo room of submarine $b$ after firing all primary task-part missiles, 0 otherwise
$D_{bw}^{BC}$ ,	number of weapons of type $w$ remain in the canister capsules of submarine $b$ after firing all primary task-part missiles, 0 otherwise
$Z_k$ ,	objective function value for goal $k \in K$
$Z_o$ ,	combined-goal objective

## B. OBJECTIVE FUNCTION

Our multi-objective function consists of the following goals:

Goal  $k = 1$ : Minimize the total number of unmet task-parts:

$$Z_1^* = \min Z_1, \text{ where } Z_1 = \sum_t \sum_p \sum_{m=1}^{n_{tp}^M} U_{tpm}$$

Goal  $k = 2$ : Minimize the use of *penalty* firing units:

$$Z_2^* = \min Z_2, \text{ where } Z_2 = \sum_s p_s^S Z_s^S + \sum_b p_b^B Z_b^B$$

Goal  $k = 3$ : Maximize the number of weapons allocated from designated *expend* firing units:

$$Z_3^* = \max Z_3, \text{ where } Z_3 = E^S + E^B$$

Goal  $k = 4$ : Minimize the deviation from average of residual number of missiles on *non-expend* firing units:

$$Z_4^* = \min Z_4, \text{ where } Z_4 = \sum_{s \in S \setminus S^{Exp}} RD_s^S + \sum_{b \in B \setminus B^{Exp}} RD_b^B$$

Goal  $k = 5$ : Maximize the number of firing units that have been assigned to allocate primary task-parts in launch areas so designated:

$$Z_5^* = \max Z_5, \text{ where } Z_5 = \sum_{a \in PR^A} \left( \sum_{s|a_s^S=a} PR_s^S + \sum_{b|a_b^B=a} PR_b^B \right)$$

Goal  $k = 6$ : Maximize the number of firing units that have been assigned to allocate back-up task-parts in launch areas so designated:

$$Z_6^* = \max Z_6, \text{ where } Z_6 = \sum_{a \in BU^A} \left( \sum_{s|a_s^S=a} BU_s^S + \sum_{b|a_b^B=a} BU_b^B \right)$$

Goal  $k = 7$ : Minimize the *missile to mission matching* ( $M^3$ ) list position of all selected weapons:

$$\begin{aligned}
Z_7^* = \min Z_7, \text{ where } Z_7 = & \sum_s \sum_{h=1}^{m_s^H} \sum_{w|n_{shw}^S > 0} \sum_{t \left| \begin{smallmatrix} s \in S_t^T, \\ w \in W_t^T \end{smallmatrix} \right.} \sum_p \sum_{m=1}^{n_{tp}^M} m_p^3 m_{tw}^3 X_{shwtpm}^S & + \\
& \sum_b \sum_{w|n_{bw}^{BT} > 0} \sum_{t \left| \begin{smallmatrix} b \in B_t^T, \\ w \in W_t^T \end{smallmatrix} \right.} \sum_p \sum_{m=1}^{n_{tp}^M} m_p^3 m_{tw}^3 X_{bwtprm}^{BT} & + \\
& \sum_b \sum_{w|n_{bw}^{BC} > 0} \sum_{t \left| \begin{smallmatrix} b \in B_t^T, \\ w \in W_t^T \end{smallmatrix} \right.} \sum_p \sum_{m=1}^{n_{tp}^M} m_p^3 m_{tw}^3 X_{bwtprm}^{BC}
\end{aligned}$$

Goal  $k = 8$ : Maximize residual salvo capability:

$$Z_8^* = \max Z_8, \text{ where } Z_8 = \sum_s \sum_{h=1}^{m_s^H} \sum_{w|n_{shw}^S > 0} v_w D_{shw}^S + \sum_b \sum_{w|n_{bw}^{BT} > 0} v_w D_{bw}^{BT} + \sum_b \sum_{w|n_{bw}^{BC} > 0} v_w D_{bw}^{BC}$$

Combined goal: Minimize a weighed average of the goals  $k \in K$  :

$$Z_0^* = \min Z_0, \text{ where } Z_0 = \sum_k -\gamma_k \omega_k z_k$$

## C. MODEL CONSTRAINTS

### 1. Constraint Formulation

- Ensure weapon availability for tasks allocated to half-modules, torpedo tubes and canister capsules:

$$\text{A.1. } \sum_{t \left| \begin{smallmatrix} s \in S_t^T, \\ w \in W_t^T \end{smallmatrix} \right.} \sum_p \sum_{m=1}^{n_{tp}^M} X_{shwtpm}^S \leq n_{shw}^S, \quad \forall s, h, w \left| 1 \leq h \leq m_s^H, n_{shw}^S > 0
\right.$$

$$\text{A.2. } \sum_{t \left| \begin{smallmatrix} b \in B_t^T, \\ w \in W_t^T \end{smallmatrix} \right.} \sum_p \sum_{m=1}^{n_{tp}^M} X_{bwtprm}^{BT} \leq n_{bw}^{BT}, \quad \forall b, w \left| n_{bw}^{BT} > 0
\right.$$

$$\text{A.3. } \sum_{t \left| \begin{smallmatrix} b \in B_t^T, \\ w \in W_t^T \end{smallmatrix} \right.} \sum_p \sum_{m=1}^{n_{tp}^M} X_{bwtprm}^{BC} \leq n_{bw}^{BC}, \quad \forall b, w \left| n_{bw}^{BC} > 0
\right.$$

- Establish the allocation at the firing unit level:

$$\text{A.4.} \quad \sum_{h=1}^{m_s^H} \sum_{w \in W_t^T \mid n_{shw}^S > 0} X_{shwtpm}^S = Y_{stpm}^S, \quad \forall s, t, p, m \mid s \in S_t^T, 1 \leq m \leq n_{tp}^M$$

$$\text{A.5.} \quad \sum_{w \in W_t^T \mid n_{bw}^{BT} > 0} X_{bwtpm}^{BT} + \sum_{w \in W_t^T \mid n_{bw}^{BC} > 0} X_{bwtpm}^{BC} = Y_{btpm}^B, \quad \forall b, t, p, m \mid b \in B_t^T, 1 \leq m \leq n_{tp}^M$$

$$\text{A.6.} \quad \sum_{s \in S_t^T} Y_{stpm}^S + \sum_{b \in B_t^T} Y_{btpm}^B \leq 1, \quad \forall t, p, m \mid 1 \leq m \leq n_{tp}^M$$

- Ensure all (or none) of the task-parts in a task are allocated:

$$\text{A.7.} \quad \sum_{s \in S_t^T} Y_{stp'm'}^S + \sum_{b \in B_t^T} Y_{btp'm'}^B = \sum_{s \in S_t^T} Y_{stpm}^S + \sum_{b \in B_t^T} Y_{btpm}^B \quad \forall t, p, m \mid 1 \leq m \leq n_{tp}^M, n_{tp'}^M > 0, \\ \text{for } p' \in \{"PR", "GH"\}, m' = "1"$$

- Prevent near-simultaneous tasks in the same half-module:

$$\text{A.8.} \quad \sum_{w \mid n_{shw}^S > 0} \sum_{t \in T_i^T \mid s \in S_t^T, w \in W_t^T} \sum_p \sum_{m=1}^{n_{tp}^M} X_{shwtpm}^S \leq 1, \quad \forall s, h, i \mid 1 \leq h \leq m_s^H$$

$$\text{A.9.} \quad \sum_{w \mid n_{bw}^{BT} > 0} \sum_{t \in T_i^T \mid b \in B_t^T, w \in W_t^T} \sum_p \sum_{m=1}^{n_{tp}^M} X_{bwtpm}^{BT} \leq m_b^{BT}, \quad \forall b, i$$

- Allocate each task-part missile requirement or take a penalty:

$$\text{A.10.} \quad \sum_{s \in S_t^T} Y_{stpm}^S + \sum_{b \in B_t^T} Y_{btpm}^B + U_{tpm} = 1, \quad \forall t, p, m \mid 1 \leq m \leq n_{tp}^M$$

- Ensure primary and back-up task-part relations:

$$\text{A.11.} \quad Y_{stpm}^S + Y_{stp'm'}^S \leq 1, \quad \forall s, t, m, m' \mid s \in S_t^T, 1 \leq m \leq n_{tp}^M, 1 \leq m' \leq n_{tp'}^M, \\ \text{for } p = "PR", p' = "BU"$$

$$\text{A.12. } Y_{btpm}^B + Y_{btp'm'}^B \leq 1, \quad \forall b, t, m, m' \mid b \in B_t^T, 1 \leq m \leq n_{tp}^M, 1 \leq m' \leq n_{tp'}^M, \\ \text{for } p = "PR", p' = "BU"$$

- Ensure primary and ready-spare task-part relations:

$$\text{A.13. } Y_{stpm}^S = Y_{stp'm}^S, \quad \forall s, t, m \mid s \in S_t^T, 1 \leq m \leq n_{tp}^M, \\ \text{for } p = "PR", p' = "RS"$$

$$\text{A.14. } Y_{btpm}^B = Y_{btp'm}^B, \quad \forall b, t, m \mid b \in B_t^T, 1 \leq m \leq n_{tp}^M, \\ \text{for } p = "PR", p' = "RS"$$

- Calculate firing unit engagement in targets and in the overall allocation:

$$\text{A.15. } Y_{stpm}^S \leq ZZ_{sg}^S, \quad \forall s, t, p, m, g \mid s \in S_t^T, 1 \leq m \leq n_{tp}^M$$

$$\text{A.16. } ZZ_{sg}^S \leq Z_s^S, \quad \forall s, g$$

$$\text{A.17. } Y_{btpm}^B \leq ZZ_{bg}^B, \quad \forall b, t, p, m, g \mid b \in B_t^T, 1 \leq m \leq n_{tp}^M$$

$$\text{A.18. } ZZ_{bg}^B \leq Z_b^B, \quad \forall b, g$$

- Limit firing units for restricted targets:

$$\text{A.19. } \sum_s ZZ_{sg}^S + \sum_b ZZ_{bg}^B \leq n_g^{FU}, \quad \forall g \in G^R$$

- Allocate all (or none) of the tasks in restricted targets:

$$\text{A.20. } U_{tpm} = U_{t'pm}, \quad \forall t, t', g \mid \begin{cases} n_{tp}^M > 0, \text{ for } p \in \{ "PR", "GH" \}, m = "1" \\ t, t' \in T_g^G, t \neq t', g \in G^R \end{cases}$$

- Use of expend firing units for primary task-parts:

$$\text{A.21. } \sum_{s \in S^{Exp}} \sum_{t \mid s \in S_t^T} \sum_{m=1}^{n_{tp}^M} Y_{stpm}^S = E^S, \quad \text{for } p = "PR"$$

$$\text{A.22. } \sum_{b \in B^{Exp}} \sum_{t \mid b \in B_t^T} \sum_{m=1}^{n_{tp}^M} Y_{btpm}^B = E^B, \quad \text{for } p = "PR"$$

- Ensure the allocation of all missiles for non-separable task-part to the same firing unit:

$$\text{A.23. } \sum_{m=1}^{n_{tp}^M} Y_{stpm}^S = n_{tp}^M NS_{stpm}^S, \quad \forall s, t, p \mid s \in S_t^T, p \in P_t^{NS}, n_{tp}^M \geq 2$$

$$\text{A.24. } \sum_{m=1}^{n_{tp}^M} Y_{btpm}^B = n_{tp}^M NS_{btpm}^B, \quad \forall b, t, p \mid b \in B_t^T, p \in P_t^{NS}, n_{tp}^M \geq 2$$

- Compute absolute deviation from average number of residual weapons onboard each firing unit:

$$\text{A.25. } \sum_{h=1}^{n_s^H} \sum_w n_{shw}^S - \sum_{t \mid s \in S_t^T} \sum_{m=1}^{n_{tp}^M} Y_{stpm}^S = RN_s^S, \quad \forall s \in S \setminus S^{Exp}, \quad \text{for } p = "PR"$$

$$\text{A.26. } \sum_w (n_{bw}^{BT} + n_{bw}^{BC}) - \sum_{t \mid b \in B_t^T} \sum_{m=1}^{n_{tp}^M} Y_{btpm}^B = RN_b^B, \quad \forall b \in B \setminus B^{Exp}, \quad \text{for } p = "PR"$$

$$\text{A.27. } \sum_{s \in S \setminus S^{Exp}} RN_s^S + \sum_{b \in B \setminus B^{Exp}} RN_b^B = n^{NExp} RAVG$$

$$\text{A.28. } RN_s^S - RAVG \geq -RD_s^S, \quad \forall s \in S \setminus S^{Exp}$$

$$\text{A.29. } RN_s^S - RAVG \leq RD_s^S, \quad \forall s \in S \setminus S^{Exp}$$

$$\text{A.30. } RN_b^B - RAVG \geq -RD_b^B, \quad \forall b \in B \setminus B^{Exp}$$

$$\text{A.31. } RN_b^B - RAVG \leq RD_b^B, \quad \forall b \in B \setminus B^{Exp}$$

- Keep track of areas that must spread primary task-part missiles:

$$\text{A.32. } \sum_{t \mid s \in S_t^T} \sum_{m=1}^{n_{tp}^M} Y_{stpm}^S \geq PR_s^S, \quad \forall s \mid a_s^S \in A^{PR}, \quad \text{for } p = "PR"$$

$$\text{A.33. } \sum_{t \mid b \in B_t^T} \sum_{m=1}^{n_{tp}^M} Y_{btpm}^B \geq PR_b^B, \quad \forall b \mid a_b^B \in A^{PR}, \quad \text{for } p = "PR"$$

- Keep track of areas that must spread back-up task-part missiles:

$$\text{A.34. } \sum_{t|s \in S_t^T} \sum_{m=1}^{n_{tp}^M} Y_{stpm}^S \geq BU_s^S, \quad \forall s | a_s^S \in A^{BU}, \quad \text{for } p = "BU"$$

$$\text{A.35. } \sum_{t|b \in B_t^T} \sum_{m=1}^{n_{tp}^M} Y_{btpm}^B \geq BU_b^B, \quad \forall b | a_b^B \in A^{BU}, \quad \text{for } p = "BU"$$

- Calculate residual salvo size:

$$\text{A.36. } n_{shw}^S - \sum_{t|s \in S_t^T, w \in W_t^T} \sum_{m=1}^{n_{tp}^M} X_{shwtpm}^S \geq D_{shw}^S, \quad \forall s, h, w | 1 \leq h \leq m_s^H, n_{shw}^S > 0, \quad \text{for } p = "PR"$$

$$\text{A.37. } \sum_{w|n_{shw}^S > 0} D_{shw}^S \leq 1, \quad \forall s, h | 1 \leq h \leq m_s^H$$

$$\text{A.38. } n_{bw}^{BT} - \sum_{t|b \in B_t^T, w \in W_t^T} \sum_{m=1}^{n_{tp}^M} X_{bwtpm}^{BT} \geq D_{bw}^{BT}; \quad p = "PR", \quad \forall b, w | n_{bw}^{BT} > 0, \quad \text{for } p = "PR"$$

$$\text{A.39. } \sum_{w|n_{bw}^{BT} > 0} D_{bw}^{BT} \leq m_b^{BT}, \quad \forall b$$

$$\text{A.40. } n_{bw}^{BC} - \sum_{t|b \in B_t^T, w \in W_t^T} \sum_{m=1}^{n_{tp}^M} X_{bwtpm}^{BC} \geq D_{bw}^{BC}; \quad p = "PR", \quad \forall b, w | n_{bw}^{BC} > 0, \quad \text{for } p = "PR"$$

- TSC specifications:

$$\text{A.41. } X_{shwtpm}^S = 1, \quad \forall (s, h, w, t, p, m) \in TSC^S$$

$$\text{A.42. } X_{bwtpm}^{BT} = 1, \quad \forall (b, w, t, p, m) \in TSC^{BT}$$

$$\text{A.43. } X_{bwtpm}^{BC} = 1, \quad \forall (b, w, t, p, m) \in TSC^{BC}$$



- Submarine back-up allocation:

$$\begin{aligned} \text{A.44. } X_{bwt\text{tpm}}^{BT} = 0, & \quad \forall b, w, t, m \left| \begin{array}{l} b \notin B^{BU}, w \in W_t^T, \\ n_{bw}^{BT} > 0, 1 \leq m \leq n_{tp}^M \end{array} \right. \\ & \quad \text{for } p = "BU", \end{aligned}$$

$$\begin{aligned} \text{A.45. } X_{bwt\text{tpm}}^{BC} = 0, & \quad \forall b, w, t, m \left| \begin{array}{l} b \notin B^{BU}, w \in W_t^T, \\ n_{bw}^{BC} > 0, 1 \leq m \leq n_{tp}^M \end{array} \right. \\ & \quad \text{for } p = "BU", \end{aligned}$$

- Force levels of achievement for goals:

$$\text{A.46. } Z_{k'} \leq Z_{k'}^*(1 + \varepsilon_k), \quad \forall k' \mid k' < k, \gamma_{k'} = -1$$

$$\text{A.47. } Z_{k'} \geq Z_{k'}^*(1 - \varepsilon_k), \quad \forall k' \mid k' < k, \gamma_{k'} = 1$$

- Binary variables:

$$\text{A.48. } X_{shw\text{tpm}}^S \in \{0, 1\}, \quad \forall s, h, w, t, p, m \mid s \in S_t^T, 1 \leq h \leq m_s^H, w \in W_t^T, t \in T, n_{shw}^S > 0, p \in P, 1 \leq m \leq n_{tp}^M$$

$$\text{A.49. } X_{bwt\text{tpm}}^{BT} \in \{0, 1\}, \quad \forall b, w, t, p, m \mid b \in B_t^T, w \in W_t^T, n_{bw}^{BT} > 0, t \in T, p \in P, 1 \leq m \leq n_{tp}^M$$

$$\text{A.50. } X_{bwt\text{tpm}}^{BC} \in \{0, 1\}, \quad \forall b, w, t, p, m \mid b \in B_t^T, w \in W_t^T, n_{bw}^{BC} > 0, t \in T, p \in P, 1 \leq m \leq n_{tp}^M$$

$$\text{A.51. } ZZ_{sg}^S \in \{0, 1\}, \quad \forall s, g \mid s \in S_t^T, g \in G$$

$$\text{A.52. } Z_s^S \in \{0, 1\}, \quad \forall s \in S_t^T$$

$$\text{A.53. } ZZ_{bg}^B \in \{0, 1\}, \quad \forall b, g \mid b \in B_t^T, g \in G$$

$$\text{A.54. } Z_b^B \in \{0, 1\}, \quad \forall b \in B_t^T$$

$$\text{A.55. } NS_{stp}^S \in \{0,1\}, \quad \forall s,t,p \mid s \in S_t^T, t \in T, p \in P$$

$$\text{A.56. } NS_{btp}^B \in \{0,1\}, \quad \forall b,t,p \mid b \in B_t^T, t \in T, p \in P$$

$$\text{A.57. } PR_s^S \in \{0,1\}, \quad \forall s \in S_t^T$$

$$\text{A.58. } PR_b^B \in \{0,1\}, \quad \forall b \in B_t^T$$

$$\text{A.59. } BU_s^S \in \{0,1\}, \quad \forall s \in S_t^T$$

$$\text{A.60. } BU_b^B \in \{0,1\}, \quad \forall b \in B_t^T$$

- Non-negative Variables (and bounds, if applicable):

$$\text{A.61. } 0 \leq Y_{spm}^S \leq 1, \quad \forall s,t,p,m \mid s \in S_t^T, t \in T, p \in P, 1 \leq m \leq n_p^M$$

$$\text{A.62. } 0 \leq Y_{btpm}^B \leq 1, \quad \forall b,t,p,m \mid b \in B_t^T, t \in T, p \in P, 1 \leq m \leq n_p^M$$

$$\text{A.63. } 0 \leq U_{tpm} \leq 1, \quad \forall t,p,m \mid t \in T, p \in P, 1 \leq m \leq n_p^M$$

$$\text{A.64. } 0 \leq D_{shw}^S \leq 1, \quad \forall s,h,w \mid s \in S_t^T, 1 \leq h \leq m_s^H, w \in W_t^T, n_{shw}^S > 0$$

$$\text{A.65. } 0 \leq D_{bw}^{BT} \leq 1, \quad \forall b,w \mid b \in B_t^T, w \in W_t^T, n_{bw}^{BT} > 0$$

$$\text{A.66. } 0 \leq D_{bw}^{BC} \leq 1, \quad \forall b,w \mid b \in B_t^T, w \in W_t^T, n_{bw}^{BC} > 0$$

$$\text{A.67. } RN_s^S \geq 0, \quad \forall s \in S_t^T$$

$$\text{A.68. } RN_b^B \geq 0, \quad \forall b \in B_t^T$$

$$\text{A.69. } RD_s^S \geq 0, \quad \forall s \in S_t^T$$

$$\text{A.70. } RD_b^B \geq 0, \quad \forall b \in B_t^T$$

$$\text{A.71. } RAVG \geq 0$$

$$\text{A.72. } E^S \geq 0$$

$$A.73. \quad E^B \geq 0$$

## 2. Multi-Objective Formulation

We consider two formulations for our multi-objective optimization problem:

1. Hierarchical formulation (H): For each  $k \in K$ , solve the following sequence of models:

$$(H_k): \quad Z_k^* = \min(\text{or max}) \quad Z_k \\ \text{s.t. (A.1) - (A.73)}$$

Remark: To be more precise, we write all these models as the following minimization problems:

$$(H_k): \quad Z_k^* = -\gamma_k \min -\gamma_k Z_k \\ \text{s.t. (A.1) - (A.73)}$$

2. Combined-goal (sometimes referred to as weighed-sum) formulation (C):

$$(C): \quad Z_o^* = \min \quad Z_o \\ \text{s.t. (A.1) - (A.45)} \\ \text{(A.48) - (A.73)}$$

## 3. Description of Formulation.

- Goals #1 ( $Z_1$ ) through #8 ( $Z_8$ ) seek to minimize (or maximize) our multiple objectives.  $Z_0$  is formulated as a linear combination of  $Z_1, \dots, Z_8$ , where the more important the goal, the higher the associated weight.
- Constraints (A.1) to (A.3) ensure that: (a) we do not exceed the available weapons of each type in any half-module on a ship, and (b) we do not exceed the available weapons of each type in the torpedo room and in the canister capsules on a submarine.
- Constraints (A.4) to (A.6) relate the half-module, torpedo room and canister capsule variables to the firing unit level. In particular, (A.4) tracks the ship (if any) used for a specific task-part mission, whereas (A.5) does the same for submarines. (A.6) ensures that each task-part missile requirement is allocated once (at most).

- Constraint (A.7) requires the allocation of either all or none of the task-part missiles in a task.
- Constraints (A.8) and (A.9) avoid allocating more than one TLAM per half-module (for ships) and more than the number of torpedo tubes available (for submarines) for task-parts associated with near-simultaneous tasks.
- Constraint (A.10) determines whether a task-part missile requirement has been allocated or not.
- Constraints (A.11) and (A.12) establish the relation between primary and back-up missions: primary and back-up task-parts for a task cannot be allocated to the same firing unit.
- Constraints (A.13) and (A.14) establish the relation between primary and ready-spare missions: primary and ready-spare task-parts for a task must be allocated to the same firing unit.
- Constraints (A.15) and (A.17) track which firing units have been used in the allocation of a specific target.
- Constraints (A.16) and (A.18) track which firing units have been used in the overall allocation.
- Constraint (A.19) limits the number of firing units assigned to a restricted target not to exceed the maximum number of firing units allowed for that target.
- Constraint (A.20) ensures that all or none of the tasks in a restricted target must be allocated.
- Constraints (A.21) and (A.22) determine if an expend firing unit is assigned to a primary task-part missile requirement.
- Constraints (A.23) and (A.24) ensure that all missile requirements for a non-separable task-part are allocated to the same firing unit.
- Constraints (A.25) to (A.31) calculate the absolute deviation from the average number of residual weapons onboard each non-expend firing unit.
- Constraints (A.32) and (A.33) keep track of the number of firing units that have been allocated primary task-parts, for areas so designated.
- Constraints (A.34) and (A.35) keep track of the number of firing units that have been allocated back-up task-parts, for areas so designated.
- Constraints (A.36) to (A.40) calculate the residual salvo-size. In particular, (A.37) ensures that only one weapon per half-module counts for computing residual salvo. (A.39) limits the maximum count for weapons in the torpedo room on a submarine to the number of torpedo tubes.
- Constraints (A.41) to (A.43) satisfy the specifications of the TSC.

- Constraints (A.44) and (A.45) ensure that back-up task-parts are not allocated to submarines that are not designated for back-up tasking.
- Constraints (A.46) and (A.47) establish levels of achievement for goals that are hierarchically superior to the goal that is being minimized (or maximized) at the incumbent iteration. (Only for the hierarchical formulation (H).)

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF REFERENCES

- Arnold, Andrew (2000). "A Fast Heuristic for Tomahawk Land-Attack Predesignation," Master's Thesis in Operations Research, Naval Postgraduate School, Monterey, California.
- Aytekin, Mehmet (2002). "Applying a Fix-and-Relax Heuristic to U.S. Navy Force Structure Planning," Master's Thesis in Operations Research, Naval Postgraduate School, Monterey, California.
- Brooke, A., Kendrick, D., Meeraus, A., and Raman, R. (1996). *GAMS: A User's Guide*.
- Brown, G., Graves, G. and Ronen, D. (1987). "Scheduling Ocean Transportation of Crude Oil," *Management Science*, Vol. 33, No. 3, pp. 335-346.
- Dillenberger, Ch., Escudero, L. F., Wollensak, A., Zhang, W. (1994). "On Practical Resource Allocation for Production Planning and Scheduling with Period Overlapping Setups," *European Journal of Operational Research*, 75, pp. 275-286.
- Escudero, L. F., Salmeron J. (2002). "On A Fix-And-Relax Framework for Large-Scale Resource-Constrained Project Scheduling," in review.
- GAMS (2003). GAMS Development Corporation, Washington, D.C. Software available from [www.gams.com](http://www.gams.com) (accessed January 2004).
- GAMS-CPLEX (2003). <http://www.gams.com/solvers/solvers.htm#CPLEX> (accessed January 2004).
- Hodge, Bertram (1999). "A Heuristic for Tomahawk Land-Attack Predesignation," Master's Thesis in Operations Research, Naval Postgraduate School, Monterey, California.
- Kirk, Brian (1999). "Tomahawk Strike Coordinator Predesignation: Optimizing Firing Platform and Weapon Allocation," Master's Thesis in Operations Research, Naval Postgraduate School, Monterey, California.
- Kubu, Justin A. (2001). "Enhancing Real-Time Tomahawk Predesignation to Diagnose Conflicts, Prescribe Improvements, and Plan Multiple Strikes," Master's Thesis in Operations Research, Naval Postgraduate School, Monterey, California.
- Kuykendall, Scott (1998). "Optimizing Selection of Tomahawk Cruise Missiles," Master's Thesis in Operations Research, Naval Postgraduate School, Monterey, California.
- Navy Office of Information (2004). <http://www.chinfo.navy.mil/navpalib/factfile/missiles/wep-toma.html> (accessed April).

Onwardoverland (2004). <http://www.onwardoverland.com/tomahawk/tomahawk.html> (accessed June).

Reynolds, Guy B., (1998). "The Nuclear-Armed Tomahawk Cruise Missile: Its Potential Utility on United States and United Kingdom Attack Submarines," Master's Thesis, Naval Postgraduate School, Monterey, California.

Rosenthal, Richard E. (1985). "Principles of Multiobjective Optimization," *Decision Sciences*, Vol. 16, 1985, pp. 133-152.

Salmeron, J., Newman, A. and Rosenthal, R. (2002). "TLAM Data and Algorithm Specification," (internal document, available upon request). Naval Postgraduate School, Monterey, California.

TLAM Documentation (2002) "Heuristic Algorithm for TLAM Predesignation Project," OR Department, Naval Postgraduate School (internal report).

TLAM Documentation (2004) "TLAM Formulations and Code," OR Department, Naval Postgraduate School (internal report).

Wingert, Paul H. (2001). "An Improvement Heuristic for Tomahawk Land-Attack Predesignation, Enhanced to Accommodate Manual Planning, and Validated with Fleet Exercise Data," Master's Thesis in Operations Research, Naval Postgraduate School, Monterey, California.

Wolsey, L. A. (1998). *Integer Programming*, John Wiley, New York.

Wu, Hsin-Fu, Newman, Alexandra M., (2002). "Incorporating the SSGN Firing Unit in the TOMAHAWK Missile Phase I Predesignation Heuristic", Naval Postgraduate School, Monterey, California, September 2002.



## INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center  
Ft. Belvoir, Virginia
2. Dudley Knox Library  
Naval Postgraduate School  
Monterey, California
3. Chief of Naval Operations (N81)  
Navy Department 2000 Navy Pentagon  
Washington, D.C.
4. Dr. Charles Fennemore  
Naval Surface Warfare Center Dahlgren Division  
Combat System Branch  
Dahlgren, Virginia
5. Kara Harp Okulu Komutanligi  
Kutuphane  
Bakanliklar, Ankara, TURKEY
6. Professor Javier Salmeron, Code OR/Sa  
Department of Operations Research  
Naval Postgraduate School  
Monterey, California
7. Professor Richard E. Roshental, Code OR/R1  
Department of Operations Research  
Naval Postgraduate School  
Monterey, California
8. Bilkent Universitesi Kutuphanesi  
Bilkent, Ankara, TURKEY
9. Orta Dogu Teknik Universitesi Kutuphanesi  
Balgat, Ankara, TURKEY
10. Bogazici Universitesi Kutuphanesi  
Bebek, Istanbul, TURKEY

11. P.Utgm. Ali DEMİR  
Kara Kuvvetleri Komutanligi  
Dag ve Komando Tugayi 3ncu Ic Guv. Tb. 2nci Ic. Guv. Bl.K.  
Hakkari, TURKEY